卒業論文

ターミナルアトラクタを組み込んだ
複製・競合メカニズムによる効率的な機械学習

Building Efficient Machine Learning
with Reproduction and Competition Mechanisms Incorporating
Terminal Attractors

富山県立大学 工学部 情報システム工学科
2120014 小澤 翔太
指導教員 奥原 浩之 教授
提出年月: 令和7年 (2025年)2月

目 次

図一覧

表一覧

記号一覧

以下に本論文において用いられる用語と記号の対応表を示す.

用語	記号
入力ニューロンの番号	j
入力ニューロン数	M
出力ニューロンの番号	i
出力ニューロン数	N
第 i ニューロンにおける微小領域の番号	k
第 i ニューロンにおける第 k 番目の微小領域	B_{ik}
内的自然増加率	α_{ik}^{j}
第 j ニューロンと第 h ニューロンの競合効果	γ_{ik}^{jh}
微小領域 B_{ik} におけるシナプス結合荷重の大きさ	w_{ik}^{j}
微小領域 B_{ik} におけるシナプス結合荷重の大きさ	w_{ik}^{j}
Positional Encoding における位置エンベディングの次元数	i
Positional Encoding における埋め込みベクトルの次元数	d_{model}
Siamese Network における埋め込み表現の次元	n
Siamese Network におけるラベルの数	k
UMAP における他の点 x_i の近傍に x_j が属する強さ	$v_{j\|i}$
UMAP における他の点 y_i の近傍に y_j が属する強さ	$w_{j\|i}$
UMAP における他の点 x_j が属する強さ	$v_{j\|i}$
UMAP における点 x_i と x_j の距離	r_{ij}
UMAP における点 y_i と y_j の距離	d_{ij}
UMAP における点 x_i に対して，k 近傍の集合	K_i
UMAP における点の疎密に対応するための変数	σ_i
k-menas における n 個の個体	$\vec{x_i} = (x_{i1}, \ldots, x_{iD})$
k-menas における n 個の個体の集合	x
k-menas における K 個の重なりの無いクラス	$X_k, k = 1, \ldots, K$
k-menas におけるクラスタの中心	$\vec{c_k}$
k-menas における $X_k^{(t)}$ に属する個体の数	$n_k^{(t)}$
k-menas における $X_k^{(t)}$ の $(K+1)$ 回目のクラスタの中心	$\vec{c_k}^{(t+1)}$
シルエット分析における各データのサンプル	$x^{(i)}$
シルエット分析における $x^{(i)}$ が属するクラスタ	C_{in}
シルエット分析における $x^{(i)}$ に最も近いクラスタ	C_{near}

第1章

はじめに

§ 1.1　本研究の背景

関数近似問題やパターン識別に適したニューラルネットワークの１つに動径基底関数ネットワーク（Radial Basis Function Network: RBFN）がある．RBFN は有限個の入出力データを補完する方法として提案された，3 層から構成されるニューラルネットワークであり，多層パーセプトロンと同じく任意の非線形関数の近似が可能であるという特長をもつ．さらに，RBFN は基底関数としてガウス関数を用いることによって階層型ニューラルネットワークに比較してニューロンごとの局所的な学習が可能であるなどの優れた点をもつ．しかし，RBFN では未知の非線形関数を近似するため，あらかじめ必要なニューロン数が不明であり冗長なニューロンを必要とする場合がある．一般に，ニューロンの増加は学習の遅延化や過学習の問題を生じることが知られている．

これらの問題を解決するために，適者生存型学習則に基づいたシナプス可塑性方程式を適用した，競合動径基底関数ネットワーク（Competitive RBFN: CRBFN）が提案されている [1]．CRBFN は，ニューロン間に競合を生じさせ，学習に必要なニューロンのみが自然に生き残るようになっており，冗長なニューロンの削減を図ることができる．しかし，CRBFN では基底関数を追加する機能は備わっておらず，基底関数の数が足りない場合は関数近似自体が不可能となる．そこで，CRBFN に基底関数を複製して追加する機能を加えた，複製・競合動径基底関数ネットワーク (Reproductive CRBFN: RC-RBFN) が提案されている [2]．

RBFN は非線形関数の線形和を用いたがけ崩れ発生限界雨量線の設定に関する研究や，複数の波源から発生された信号の到来方向の予測など非線形関数の近似器として多岐にわたって活用されている [3][4]．

§ 1.2　本研究の目的

特許情報は，技術開発の成果を客観的に反映した貴重な情報源であり，技術動向の把握や競合他社の分析など，様々な場面で活用されている．しかしながら，近年の技術開発の加速とグローバル化に伴い，世界的な特許出願件数が急激に増加している．世界知的所有権機関 (WIPO) の統計によると，2021 年の世界の特許出願件数は約 340 万件にのぼり，前年比 3.6%増加した．2022 年も世界の特許出願件数は前年比 1.7%増の約 346 万件となり，過去最高を 2 年連続で更新した．2010 年時点で約 199 万件であった世界の特許出願件数は，この 10 年間で 1.6 倍以上に急増し，2019 年には約 322 万件に達している [?]．

一方で，情報処理技術の発展に伴い，コンピューターが人間の創造的な問題解決や思考活動を支援する発想支援システムの研究が進展している [?]．今後の時代においては，より多様なアイデアを発想する能力が重要視されると考えられている．人間が創造活動を行う際，自分の考えを言語で整理し修正を行うことが多いことがわかっている．認知心理学においても思考と言語の深い関係が指摘されており，言葉を通じた表現と共有が創意工夫を促進する効果があると考えられている．したがって，人工知能が発想支援を行うためには，人間の言葉を理解する必要がある．しかし，人間の自然言語は複雑で，その内容を正しく理解することは極めて困難であり，機械独自の自然言語処理手法が用いられている．最近では sentence-BERT などのディープラーニングを用いた自然言語処理技術が進展しており，今後も自然言語処理技術の進展が重要視されている．

このように特許出願件数が急増する中で，膨大となった特許情報を人の手のみで処理し切れない状況となっている．こうした課題に対処するため，大量の特許文書を自動処理できる人工知能技術への期待が高まっている．さらに，特許の調査によれば産業界における IP ランドスケープの必要性は 8 割以上が認識しているものの，実際に IP ランドスケープを十分に実施できている企業は 1 割程度にとどまるという調査結果もある [?]．多くの企業で，IP ランドスケープの重要性は理解しつつも，実践面でのハードルがある状況である．その原因の一つとして，特許データの大規模分析に適したツールの不足があげられる．現在の特許プラットフォームでは個別の特許情報の参照には適しているが，特許全体をビッグデータとして分析を行いたい場合には適しているとは言い難い．

そこで本研究では，今日に至るまで蓄積された膨大な特許文書群を対象とした知識発見を目的とする．過去から現在に至る技術開発の流れを可視化し，その中から新たな知見を抽出することを通じて，技術動向把握や新規アイデア創出の支援を目指す．特に，過去の特許から最近の特許までを網羅的に分析し，技術の系譜や将来の発展方向性を俯瞰的に捉えることで，研究開発戦略立案への寄与を図る．

分析に際しては，特許文書に対してテキストマイニングやトピックモデリングといった自然言語処理技術を適用し，技術間の関係性把握や技術トレンドの変遷の定量化を実現する．得られた知見をインタラクティブな視覚情報で提示することで，ユーザーが直感的に技術動向を探索できる仕組みを構築する．こうしたアプローチによって，限られた人的リソースで，複雑化する技術環境を効率的かつ戦略的に把握できることが期待される．

§ 1.3　本論文の概要

本論文は次のように構成される．

第 1 章 本研究の背景と目的について説明する．

第 2 章 IP ランドスケープの概要と，特許情報処理及びそれらに用いる自然言語処理の手法についてまとめる．

第 3 章 特許文章群をベクトル化し，それらを可視化する手法についてまとめる．

第 4 章 提案手法について説明する．

第 5 章 実際の事例を設けて，第 4 章で述べた手法で，IP ランドスケープ実施の支援を行い，システムの評価を行う．

第 6 章 本論文における前章までの内容をまとめつつ，本研究で実現できたことと今後の展望について述べる．

第2章

複製・競合を考慮した動径基底関数ネットワーク

§ 2.1　競合動径基底関数ネットワーク

ニューラルネットワークは大きく分けて，素子であるニューロン，それらを結合するシナプス，そして動作規則により構成される．なかでも，記憶にもっとも関係した情報処理は，シナプスにおいて行われているとされる．記憶には種々のものが考えられるが，本研究では短期記憶と長期記憶に着目し，短期記憶はニューロンの発火頻度，長期記憶は細胞膜の特性の変化により生じるものとする．シナプスの可塑性を記述する方程式は，これらの要因を含んだものとなっていなければならない．さらに，微小な領域では成長や活動に必要な神経成長因子（Nerve Growth Factor: NGF）は競合によってシナプスに摂取される．これらの事実もシナプス可塑性のモデル化において重要な要因であると考えられる．

そこで，発火頻度や膜の特性変化を生じる物質の時間変化と，微小な領域での競合を考慮したシナプス結合荷重の大きさの時間変化は以下の方程式に従うものとする．

$$\frac{dw_{ik}^{j}}{dt} = \alpha_{ik}^{j} w_{ik}^{j} + g_{ik}^{j} w_{ik}^{j} + f_{ik}^{j} \tag{2.1}$$

$w_{ik}^{j} \geq 0$ である．また，g_{ik}^{j} は微小領域 B_{ik} に供給される NGF のうち，その領域に付着している第 j ニューロンのシナプスが入手できる量であり，f_{ik}^{j} は NGF と環境因子に依存するゆらぎである．α_{ik}^{j} は内的自然増加率であり，Hebb 則を表す．内的自然増加率 α_{ik}^{j} は以下のように定義される．

$$\alpha_{ik}^{j} = \int_{x \in B_{ik}} \eta_{ik}(x) \xi_{ik}^{j}(x) dx \tag{2.2}$$

ここで，RBFN は非線形関数 $\eta_{ik}(x)$ を動径基底関数 $\xi_{ik}^{j}(x)$ の足し合せで近似するニューラルネットワークであり，動径基底関数としては規格化されたガウス型活性関数などが用いられる．第 j ニューロン $(j = 1, 2, ..., M)$ はパラメータ m_j，σ_j をもつ．第 j ニューロンは入力に対して

$$\xi_{ik}^{j}(x) = exp\{-\frac{(x - m_j)^2}{2\sigma_j^2}\} \tag{2.3}$$

を出力する．

ここで，NGF の量 g_{ik}^{j} は次の方程式に従う．

$$\frac{dg_{ik}^{j}}{dt} = \epsilon_{ik}^{j}(G_{ik} - g_{ik}^{j}) - (\beta_{ik}^{j} w_{ik}^{j} + \sum_{h \neq j} \beta_{ik}^{h} w_{ik}^{h})$$
$$= \epsilon_{ik}^{j}(G_{ik} - g_{ik}^{j}) - \sum_{h} \beta_{ik}^{h} w_{ik}^{h} \tag{2.4}$$

G_{ik} は B_{ik} への NGF の供給速度であり，膜の特性により決定される変数である．ϵ_{ik}^{j}，β_{ik}^{h} は正の定数である．NGF の量の時間変化もシナプスの興奮性，抑制性によらず，そのシナプス間感度の大きさに依存する．また，領域へ付着するシナプスが入手し得る NGF の量の時間変化に対し，NGF の供給速度の時間変化が無視できるとして G_{ik} を定数とみなす．シナプス間感度の時間変化は発火頻度に依存するため，シナプス間感度の時間変化に対し，NGF の量の時間変化は無視できるものとする．そこで，隷従化原理を適用することにより，第 j ニューロンと第 h ニューロンが同時に NGF を消費することによる競合の効果 γ_{ik}^{jh} を導入すると以下のシナプス可塑性方程式が導かれる．

$$\frac{dw_{ik}^{j}}{dt} = (G_{ik} + \alpha_{ik}^{j} - \frac{1}{\epsilon_{ik}^{j}} \sum_{h} \beta_{ik}^{h} w_{ik}^{h}) w_{ik}^{j} + f_{ik}^{j}$$
$$= (G_{ik} + \alpha_{ik}^{j} - \sum_{h} \gamma_{ik}^{jh} w_{ik}^{h}) w_{ik}^{j} + f_{ik}^{j} \tag{2.5}$$

ここでは，競合係数 γ_{ik}^{jh} を

$$\gamma_{ik}^{jh} = \frac{\beta_{ik}^{h}}{\epsilon_{ik}^{j}} = \int_{x \in B_{ik}} \xi_{ik}^{j}(x) \xi_{ik}^{h}(x) dx \tag{2.6}$$

で定義する．

簡単のため，以下では微小領域 B_{ik} に着目することで添え字の i と k を省略し，NGF の摂取量が一定 $(G_{ik} = 0)$ で揺らぎのない場合 $(f_{ik}^{j} = 0)$ を考える．このとき，シナプス可塑性方程式は

$$\frac{dw^{j}}{dt} = (\alpha^{j} - \sum_{h} \gamma^{jh} w_{ik}^{h}) w^{j} \tag{2.7}$$

であり，正定関数 $V(\mathbf{w})$ として

$$V(\mathbf{w}) = \frac{1}{2} \int_{x \in B_{ik}} \{\eta(x) - s(x)\}^2 dx \tag{2.8}$$

を定義する．ここで，$\mathbf{w} \equiv [w^1, w^2, ..., w^M]$ であり，

$$s(x) = \sum_{j=1}^{M} w^{j} \xi^{j}(x) \tag{2.9}$$

である．この式の右辺は微小領域 B_{ik} に付着しているすべてのシナプス前終末のシナプス前発火頻度 $\xi^{j}(x)$ と，シナプス結合荷重 w^{j} との積の総和であるので，$s(x)$ を神経伝達物

質放出量と呼ぶこととする．正定関数 $V(\mathbf{w})$ はシナプス後発火頻度 $\eta(x)$ と神経伝達物質放出量 $s(x)$ の差を表す指標である．シナプス後発火頻度 $\eta(x)$ が時間に依存しないと仮定すると，その時間変化が

$$\begin{aligned}\frac{dV(\mathbf{w})}{dt} &= \sum_{j=1}^{M}\frac{\partial V(\mathbf{w})}{\partial w^j}\frac{dw^j}{dt}\\ &= -\sum_{j=1}^{M}[\int_{x\in B_{ik}}\eta(x)\xi^j(x)dx - \sum_{h=1}^{M}\int_{x\in B_{ik}}\xi^j(x)\xi^h(x)dxw^h]\frac{dw^j}{dt}\\ &= -\sum_{j=1}^{M}w^j(\alpha^j - \sum_{h=1}^{M}\gamma^{jh}w^h)^2\\ &\leq 0 \end{aligned} \tag{2.10}$$

となるため，正定関数 $V(\mathbf{w})$ が Lyapunov 関数となることがわかる．

これらから，シナプス後発火頻度 $\eta(x)$ を入力 x に対する望ましい出力，シナプス前発火頻度 $\xi^j(x)$ を動径基底関数であるとみなすことで，シナプス結合荷重 w^j は競合を行いながら RBFN と同様に望ましい出力と動径基底関数の 2 乗誤差関数を減少させることが示される．本論文では，(2.1) 式のシナプス可塑性方程式を適者生存型学習則ということとする．

次に，これを学習則として適用した動径基底関数ネットワークを提案する．動径基底関数ネットワークは階層型ニューラルネットワークに比較してニューロンごとの局所的な学習が可能であるなどの優れた点をもつため，関数近似問題やパターン識別に適用され成果を上げている．しかし，動径基底関数ネットワークでは未知の非線形関数を近似するためにあらかじめ必要なニューロン数が不明であるために冗長なニューロンを必要とする．一般に，ニューロンの増加は学習の遅延化や過学習の問題を生じることが知られている．そこで，冗長なニューロンを削除する機能を備えた CRBFN が提案されている．CRBFN ではシナプス結合荷重間に競合を生じさせ，学習に必要なニューロンのみが自然に生き残り，学習の効率化を図ることができる．

RBFN による関数近似は 2 乗誤差評価関数

$$E(\mathbf{w}) = \frac{1}{2}\sum_{j}^{M}\{\eta(\mathbf{x}_j) - s(\mathbf{x}_j)\}^2 \tag{2.11}$$

を減少させることにより実現される．つまり，RBFN が学習により獲得しなければならないのは，第 j ニューロンのシナプス結合荷重 w^j，パラメータ m_j ならびにパラメータ σ_j である．学習アルゴリズムに Delta ルールを適用することで

$$\begin{aligned}\frac{dw^j}{dt} &= -\Delta\frac{\partial E(\mathbf{w})}{\partial w^j}\\ &= \Delta\sum_{x}\{\eta(x) - s(x)\}\xi^j w^j(x),\\ \frac{dm_j}{dt} &= -\Delta\frac{\partial E(\mathbf{w})}{\partial m_j}\end{aligned} \tag{2.12}$$

$$= \Delta \sum_x \{\eta(x) - s(x)\} w^j \xi^j(x) \frac{(x - m_j)}{{\sigma_j}^2}, \tag{2.13}$$

$$\frac{d\sigma_j}{dt} = -\Delta \frac{\partial E(\mathbf{w})}{\partial \sigma_j}$$

$$= \Delta \sum_x \{\eta(x) - s(x)\} w^j \xi^j(x) \frac{(x - m_j)^2}{{\sigma_j}^3} \tag{2.14}$$

が得られる．Δ は適当な正の定数である．このような学習則に従う RBFN を CRBFN とする．

§ 2.2　ターミナルアトラクタ

シナプス可塑性方程式に従うシナプス結合荷重では，競合に負けたシナプス結合荷重は平衡状態では 0 になる．ところが，平衡解への漸近は指数関数的に行われるので原理的には有限時間で平衡状態へ到達することはできない．そこで，望ましい出力が動径基底関数を定数倍して足し合わせることで実現できる特別の場合に，あらかじめ与えられた時刻 t^* で平衡解へ収束できるように修正されたシナプス可塑性方程式を導出する．

まず，望ましい時刻 t^* で収束するシナプス結合荷重の時間変化を Lyapunov 関数を用いて規定する．そこで，Lyapunov 関数の時間変化を

$$\frac{dV(\mathbf{w})}{dt} = -\frac{V(\mathbf{w}^0)^R V(\mathbf{w})^{\frac{1}{r}}}{Rt^*} \tag{2.15}$$

で定義する．ここで，r は任意の奇数であり，$R = \frac{(r-1)}{r}$ である．$\mathbf{w}^0$ は $\mathbf{w}$ の初期値である．このような定義が可能となったのは，シナプス可塑性方程式に対する Lyapunov 関数が導出され，望ましい出力が動径基底関数を定数倍して足し合わせることで実現できる特別の場合を考えているからである．シナプス可塑性方程式は

$$\frac{dw^j}{dt} = \Delta(\alpha^j - \sum_{h=1}^{M} \gamma^{jh} w^h) w^j \tag{2.16}$$

とすることができる．このとき，Lyapunov 関数の時間変化は

$$\frac{dV(\mathbf{w})}{dt} = -\Delta \sum_{j=1}^{M} w^j (\alpha^j - \sum_{h=1}^{M} \gamma^{jh} w^h)^2 \tag{2.17}$$

となる．ここで，Δ は式 (2.15) と式 (2.17) から求めることができ，

$$\Delta = \frac{1}{\sum_{j=1}^{M} w^j (\alpha^j - \sum_{h=1}^{M} \gamma^{jh} w^h)^2} \times \frac{V(\mathbf{w}^0)^R V(\mathbf{w})^{\frac{1}{r}}}{Rt^*} \tag{2.18}$$

と決定される．以上のことから，望ましい時刻 t^* で平衡解へ収束する修正されたシナプス可塑性方程式を

$$\frac{dw^j}{dt} = \frac{(\alpha^j - \sum_{h=1}^{M} \gamma^{jh} w^h) w^j}{\sum_{j=1}^{M} w^j (\alpha^j - \sum_{h=1}^{M} \gamma^{jh} w^h)^2} \times \frac{V(\mathbf{w}^0)^R V(\mathbf{w})^{\frac{1}{r}}}{Rt^*} \tag{2.19}$$

で定義することができる．

§ 2.3 基底関数の複製

2.2 節までで冗長なニューロンを削除する手法を提示してきた．これに対し，N.N. に関数近似に必要な数のニューロンが存在しない場合は，関数近似をすること自体が不可能となる．そこで，新たに必要なニューロンを追加する手法が提案されている．これら従来の研究には，しきい値などを考え動径基底関数の削除と追加を行うものもある．ところが，このような手法では基準となるしきい値の決定自体が困難であることが予想できる．また，教師信号が動的に変化する環境では削除する手法と追加する手法を組み合わせて学習を行わなければならない．当然それぞれにとって良い手法を，ただそのまま組み合わせただけでは，動径基底関数の数が振動するなどして望ましい結果が得られるとは限らない．

そこで本研究では，まず新しい動径基底関数を追加する手法を提案する．この手法は必要な動径基底関数を効率的に追加することができる．そして，先に提案した CRBFN にこの手法を組み合わせたニューラルネットワークとして複製・競合動径基底関数ネットワーク（Reproductive CRBFN：以下，RC-RBFN）を提案する．この RC-RBFN は，環境の変化に適応する能力を備えたものとなっている．

まず，パラメータが従う確率密度関数の導出を行う．ここでは，CRBFN のシナプス結合荷重と平均ベクトル，標準偏差が学習終了時にとる同時確率密度 $p(\mathbf{w}, \mathbf{m}, \Sigma)$ を導出する．ここで，$\mathbf{m} \equiv [m_1, m_2, ..., m_M], \Sigma \equiv [\sigma_1, \sigma_2, ..., \sigma_M]$ である．シナプス結合荷重 w^j を

$$y^{j\,2} = w^j \tag{2.20}$$

と変数変換する．y^j の定義域は任意の実数である．このとき，(2.7) は

$$\frac{dy^j}{dt} = (\frac{\alpha^j}{2} - \sum_{h=1}^{M} \frac{\gamma^{jh}}{2} y^{h\,2}) y^j \tag{2.21}$$

となる．(2.20) は積分条件

$$\frac{\partial}{\partial y^h} \frac{dy^j}{dt} = \frac{\partial}{\partial y^j} \frac{dy^h}{dt} \tag{2.22}$$

を満たすため，ポテンシャル

$$U'(\mathbf{y}, \phi) = -\sum_{j=1}^{M} \int_a^{y^j} (\frac{\alpha^j}{2} - \sum_{h=1}^{M} \frac{\gamma^{jh}}{2}) y^{j\prime} dy^{j\prime} \tag{2.23}$$

を考えることができる．ただし $\mathbf{y} \equiv [y^1, y^2, ..., y^M]$ であり，パラメータ ϕ_j は平均ベクトルと共分散行列の集合 $\{\mathbf{m}_j, \Sigma_j\}$ である．そして ϕ で集合 $\{\phi_1, \phi_2, ..., \phi_M\}$ を表す．変数 y^j の時間変化はポテンシャル $U'(\mathbf{y}, \phi_j)$ から，

$$\frac{dy^j}{dt} = -\frac{\partial U'(\mathbf{y},\phi)}{\partial y^j} \tag{2.24}$$

で導くことができ，関係式 (2.19) からポテンシャル $U'(\mathbf{y},\phi)$ は

$$U(\mathbf{y},\phi) = -\sum_{j=1}^{M}\{\frac{\alpha^j}{4} - \sum_{h\neq j}^{M}\frac{\gamma^{jh}}{4}w^j w^h - \frac{\gamma^{jj}}{8}{w^j}^2\} \tag{2.25}$$

と書き直すことができる．この結果,

$$E(\mathbf{w},\phi) = 4U(\mathbf{w},\phi) + \frac{1}{2}\sum_{i}^{N}\eta^2(x_i) \tag{2.26}$$

であることが示される．ただし，$\mathbf{x}_i \equiv [x_1, x_2, ..., x_N]$ である．よって，累積 2 乗誤差関数 $E(\mathbf{w},\phi)$ の最小化はポテンシャル $U(\mathbf{w},\phi)$ の最小化と等価であることがわかる．

今，(2.20) に従う y^j はポテンシャル $U'(\mathbf{w},\phi)$ の最急降下方向に更新される．その結果，ひとたび極小解に収束すると，そこから逃れることができなくなる．そこで，極小解から脱出させるための手法として，y^j の更新則を

$$y^j(t+\Delta t) = y^j(t) - \frac{\partial U(\mathbf{w},\phi)}{\partial y^j}\delta t + \sqrt{Q\Delta t}n_j(t) \tag{2.27}$$

のようにノイズを考慮し離散近似した見本過程で与えることが考えられる．ただし，$n_j(t)$ は独立な確率変数であり，平均 0，分散 1 の正規分布 $N(0,1)$ に従う．Q は任意の正の定数である．このとき，学習終了時に CRBFN のシナプス結合荷重と平均ベクトル，標準偏差が満たす同時確率密度 $p_\beta(\mathbf{w},\phi)$ は

$$p_\beta(\mathbf{w},\phi) = Z_\beta^{-1}exp\{-\beta U(\mathbf{w},\phi)\} \tag{2.28}$$

で得ることができる．ここで $\beta = 2/Q$ である．Z_β は分配関数であり

$$Z_\beta = \int_{\mathbf{w}}\int_{\phi} exp\{-\beta V(\mathbf{w},\phi)\}d\mathbf{w}d\phi \tag{2.29}$$

で定義される．また，式 (21) はポテンシャル $V(\mathbf{w})$ と累積 2 乗誤差関数 $E(\mathbf{w})$ の関係式 (2.26) より

$$p_{\beta'}(\mathbf{w}) = Z_{\beta'}^{-1}exp\{-\beta' E(\mathbf{w})\} \tag{2.30}$$

と書き直すことができる．ここで，$\beta' = (2Q)^{-1}$ である．また，$Z_{\beta'}$ は分配関数である．

以上のようにして，パラメータが従う確率密度関数が導出できたことにより，与えられた条件のもとで累積 2 乗誤差関数 $E(\mathbf{w})$ を最小とするパラメータの値が検出できることを示す．ここでは，教師信号 $\eta(x)$ を

$$\eta(x) = 3N(-1.5,1) + 2N(1,0.5) \tag{2.31}$$

で与えることとする．$N(m,\Sigma)$ は平均 m，分散 Σ の正規密度関数を表す．この教師信号を動径基底関数を一つ（シナプス結合荷重 $w = 1$，パラメータ $\Sigma = 0.2$）だけ用いて近似す

ることを考える．この場合，近似しようとしている非線形関数 $\eta(x)$ の複雑さに対し，必要とされる動径基底関数が十分に存在していないため，累積 2 乗誤差関数 $E(\mathbf{w})$ を 0 にすること自体が不可能である．しかし，この動径基底関数のパラメータ m が従う条件付き確率密度関数

$$p_{\beta'}(m|w,\Sigma) = \frac{p_{\beta'}(w)}{\int_m p_{\beta'}(w)dm} \tag{2.32}$$

は導出することができる．

よって，シナプス結合荷重 $w = 1$，パラメータ $\Sigma = 0.2$ をもつ動径基底関数が与えられた条件のもとで累積 2 乗誤差関数 $E(\mathbf{w},\phi)$ を最小とするためには，パラメータ m を条件付き確率 $p_{\beta'}(m|w,\Sigma)$ を最大とする値に定めればよいことがわかる．また，もし同じ形質（パラメータ $w = 1$，$\Sigma = 0.2$）をもつ動径基底関数を一つ追加することができるなら，条件付き確率 $p_{\beta'}(m|w,\Sigma)$ を極大とするパラメータ m へ配置することが最も累積 2 乗誤差関数 $E(\mathbf{w},\phi)$ を小さくできることもわかる．

4. 1 自由エネルギーの導出

一般に，式 (2.13) に従いパラメータ m_j を更新し続けると極小解にとらわれ，累積 2 乗誤差関数 $E(\mathbf{w},\phi)$ の値を 0 にすることができないことがある．または，近似しようとしている非線形関数 $\eta(x)$ の複雑さに対し，必要とされる動径基底関数が十分に存在していないときには，2 乗誤差関数 $E(\mathbf{w},\phi)$ の値を 0 にすること自体が不可能である．

ところで，確率的な要素や未知の教師信号などが存在しないものとするなら，すべての入力ベクトル $\mathbf{x}_i$ ごとに動径基底関数を作成し，シナプス結合荷重が $w^i = \eta(\mathbf{x}_i)$ かつパラメータ $\Sigma_i \to 0$ であるときに，パラメータ $\mathbf{m}_i$ が $\mathbf{x}_i$ となることで近似的に 0 とできる場合がある．ここで，0 は零行列を表す．もちろん，多くの問題ではすべての入力ベクトルについて動径基底関数を用意しなくても，このようなことが可能であるものと思われる．そこで本研究では，累積 2 乗誤差関数 $E(\mathbf{w},\phi)$ の値がある正数 $\epsilon > 0$ より大きな値に収束し，学習が収束したと判断されるときに，新たに必要な動径基底関数を追加する手法を提案する．ここで提案する手法では，前章で導出した確率密度関数を利用しているため，学習が収束した時点で得られている動径基底関数の一部の形質（シナプス結合荷重 w_j，パラメータ Σ_j）が新たに追加される動径基底関数がもつパラメータに引き継がれている．そのため，効率的に最も累積 2 乗誤差関数を小さくするパラメータ $\mathbf{m}$ に動径底関数を追加していくことができる．なおかつ，最悪の場合にはすべての入力ベクトル $\mathbf{x}_i$ をパラメータ $\mathbf{m}_i$ とする動径基底関数を作成することができる．

ところで，累積 2 乗誤差関数 $E(\mathbf{w},\phi)$ の最小化は各入力ベクトル $\mathbf{x}_i$ ごとに 2 乗誤差関数 $E(\mathbf{x}_i,\mathbf{w},\phi)$ を最小化することに等価である．そこで，各入力ベクトル $\mathbf{x}_i$ に依存した平均ベクトル $\mathbf{m}_{j[i]}$ を考える．そして，学習収束の時点で得られている第 j 番目の動径基底関数に着目すると，入力ベクトル $\mathbf{x}_i$ の条件付き確率密度関数は

$$p_{\beta'}(\mathbf{x}_i|\mathbf{m}_{j[i]},\phi'_j,\phi''_j) = Z^{-1}_{\beta'}(\mathbf{m}_j,\phi'_j,\phi''_j)exp\{-\beta' E(\mathbf{x}_i,\mathbf{m}_{j[i]},\phi'_j,\phi''_j)\} \tag{2.33}$$

と導出できる．ここで，パラメータ ϕ'_j は着目した第 j 番目の動径基底関数のシナプス結合荷重 w_j と共分散行列 Σ_j の集合であり，パラメータ ϕ''_j は着目した第 j 番目の動径基底関数以外のシナプス結合荷重，共分散行列並びに平均ベクトルの集合である．以後は記法の

簡便のため，パラメータ ϕ'_j とパラメータ ϕ''_j は省略する．また，分配関数は

$$Z_{\beta'}(\mathbf{m}_j) = \sum_{i=1}^{N} exp\{-\beta' E(\mathbf{x}_i, \mathbf{m}_{j[i]})\} \tag{2.34}$$

で定義される．

条件付き確率密度関数 $p_{\beta'}(\mathbf{x}_i|\mathbf{m}_{j[i]})$ は，確率の正規化と 2 乗誤差関数 $E(\mathbf{x}_i, \mathbf{m}_{j[i]})$ の条件付き期待値

$$\langle E(\mathbf{m}_j)\rangle_{\beta'} = \sum_{i=1}^{N} p_{\beta'}(\mathbf{x}_i|\mathbf{m}_{j[i]}) E(\mathbf{x}_i, \mathbf{m}_{j[i]}) \tag{2.35}$$

が一定となるという二つの制約のもとで，エントロピー

$$\langle E(\mathbf{m}_j)\rangle_{\beta'} = \sum_{i=1}^{N} p_{\beta'}(\mathbf{x}_i|\mathbf{m}_{j[i]}) E(\mathbf{x}_i, \mathbf{m}_{j[i]}) \tag{2.36}$$

を最大にする確率密度関数として導出できる．ここで，記号

$\langle\cdots\rangle_{\beta'}$ は $p_{\beta'}(\mathbf{x}_i|\mathbf{m}_{j[i]})$ を掛けて $\mathbf{x}_i$ に関する和をとる演算を表すものとする．このとき，自由エネルギーを

$$F_{\beta'}(\mathbf{m}_j) = -\frac{1}{\beta'} \log Z_{\beta'}(\mathbf{m}_j) \tag{2.37}$$

で定義すれば,

$$S_{\beta'}(\mathbf{m}_j) = -F_{\beta'}(\mathbf{m}_j) + \beta' \langle E(\mathbf{m}_j)\rangle_{\beta'} \tag{2.38}$$

と表すことができる．この式はエントロピー $S_{\beta'}(\mathbf{m}_j)$ を最大化する条件付き確率密度関数 $p_{\beta'}(\mathbf{x}_i|\mathbf{m}_{j[i]})$ は，自由エネルギー $F_{\beta'}(\mathbf{m}_j)$ を最小化するものであることを示している．このような自由エネルギーは，データのクラスタリングのための手法であるメルティングにおいても同様に定義されている．メルティングとは，$m_{j[i]} = \mathbf{x}_i (i = 1, 2, ...N)$ かつ β' が ∞ である初期状態から，徐々に β' を 0 へ近づけていきながら，パラメータ $m_{j[i]}$ を自由エネルギー $F_{\beta'}(\mathbf{m}_j)$ の最急降下方向に更新していくものである．その結果，パラメータ $m_{j[i]}$ は徐々に同じ値をとりはじめ，最終的に一つの値 $m_{j[i]} = m_j(\forall i)$ に収束する．

4. 2 複製する位置の決定法

そこで，RC-RBFN ではパラメータ m_j の更新則を式 (2.13) の Δm_j の代わりに

$$\begin{aligned}
\Delta m_{\beta'} &= -\epsilon \sum_{i=1}^{N} \frac{\partial F_{\beta'}(\mathbf{m}_j)}{\partial m^k_{j[i]}} \\
&= -\epsilon \sum_{i=1}^{N} \frac{\partial F_{\beta'}(\mathbf{m}_j)}{\partial Z_{\beta'}(\mathbf{m}_j)} \frac{\partial Z_{\beta'}(\mathbf{m}_j)}{\partial m^k_{j[i]}} \\
&= \sum_{i=1}^{N} p_{\beta'}(\mathbf{x}_i|\mathbf{m}_{j[i]}) \Delta m_{j[i]}
\end{aligned}$$

$$= \langle \Delta m_j \rangle_{\beta'} \tag{2.39}$$

で与えることとする．ここで，

$$\Delta m_{j[i]} = -\epsilon \frac{\partial E(\mathbf{x}_i, \mathbf{m}_{j[i]})}{\partial m_{j[i]}} \tag{2.40}$$

である．

特に $\beta' = 0$ であり，初期の状態が $m_{j[i]} = m_j(\forall i)$ である場合は

$$\Delta_0 m_j = \Delta m_j \tag{2.41}$$

であることが示される．この場合は，RC-RBFN のパラメータ m_j の更新則が従来の RBFN のパラメータ m_j の更新則そのものとなっていることがわかる．このとき，$\beta' = 0$ で固定したままパラメータを $\Sigma_j \to 0$ にすると，$\Delta_0 m_j = 0$ とするパラメータ $m_{j[i]}$ は

$$\sum_{i=1}^{N} \xi(\mathbf{x}_i, \mathbf{m}_{j[i]})(\mathbf{x}_i - \mathbf{m}_{j[i]})\{\eta(x_i) - s(\mathbf{x}_i, \mathbf{m}_{j[i]})\} = 0 \tag{2.42}$$

を満たし，$\mathbf{m}_{j[i]} = \mathbf{x}_i(\forall i)$ であることがわかる．つまり，教師入力信号がパラメータ $\mathbf{m}_j$ の収束点として検出されることとなる．逆に，パラメータ Σ_j を固定したまま $\beta' \to \infty$ にすると，$\Delta_\infty m_j^k = 0$ とするパラメータ $m_{j[i]}$ は

$$\sum_{i=1}^{N} exp\{-\beta' E(\mathbf{x}_i, \mathbf{m}_{j[i]})\}\xi(\mathbf{x}_i, \mathbf{m}_{j[i]})(\mathbf{x}_i - \mathbf{m}_{j[i]})\{\eta(\mathbf{x}_i) - s(\mathbf{x}_i, \mathbf{m}_{j[i]})\} = 0 \tag{2.43}$$

を満たし，$\mathbf{x}_i(\forall i)$ を含む任意の値となることがわかる．これらの結果から，提案する RC-RBFN のパラメータ m_j の更新則 $\Delta_{\beta'} m_j$ では，$\Delta_0 m_j$ で従来の RBFN のパラメータ m_j の更新則を実現し，更に，$\Delta_\infty m_j^k$ とすることで，すべての入力ベクトル $\mathbf{x}_i$ の第 k 要素 $x_i^k(\forall i)$ を含む任意の値を安定な収束点とすることができる．ここで，提案手法とゆう度解析との関係について示す．まず，次のような自由エネルギー

$$F_{\beta'} = -\frac{1}{\beta'} \log Z_{\beta'} \tag{2.44}$$

を考える．ただし，分配関数は

$$Z_{\beta'} = \sum_{i=1}^{N} \sum_{j=1}^{M} exp\{-\beta' E(\mathbf{x}_i, \mathbf{m}_{j[i]})\} \tag{2.45}$$

で与えられる定数である．このとき，式 (2.47) は

$$\begin{aligned} F_{\beta'}(\mathbf{m}_j) - F_{\beta'} &= -\frac{1}{\beta'} \log \frac{Z_{\beta'}(\mathbf{m}_j)}{Z_{\beta'}} \\ &= -\frac{1}{\beta'} \log \sum_{i=1}^{N} p_{\beta'}(\mathbf{x}_i, m_{j[i]}) \end{aligned} \tag{2.46}$$

に変形することができる．ここで，確率密度関数 $p_{\beta'}(\mathbf{x}_i|\mathbf{m}_{j[i]})$ は

$$p_{\beta'}(\mathbf{x}_i|\mathbf{m}_{j[i]}) = Z_{\beta'}^{-1} exp\{-\beta' E(\mathbf{x}_i, \mathbf{m}_{j[i]})\} \tag{2.47}$$

である．つまり，自由エネルギー $F_{\beta'}(\mathbf{m}_j)$ のパラメータ m_j^k に関する最小化は，対数ゆう度関数に関する最大化に等価であることを示すことができる．このような対数ゆう度関数と自由エネルギーとの関係は，EM アルゴリズムに関しては既に詳しい議論がされている．

以上のことから，動径基底関数の複製を考慮した RC-RBFN の学習則を次のように提案する．

［RC-RBFN の学習則］

STEP 1. シナプス結合荷重 w_j を式 (8) のシナプス可塑性方程式により更新，パラメータ m_j を式 (32) の $\Delta_0 m_j$ により更新，パラメータ Σ_j は式 (7) により更新する．

STEP 2. 累積 2 乗誤差関数が $E(\mathbf{w}, \phi) \neq 0$ となったら学習終了．ある正数 $\epsilon > 0$ より大きな値に収束したなら STEP 3. へいく．

STEP 3. 学習収束時に得られているすべての動径基底関数について，β' を 0 から徐々に大きくしていきながら，式 (32) に従いパラメータ m_j を更新する．

STEP 4. 分岐により $\Delta_{\beta'} m_j = 0$ となる点が増えたとき，第 j 動径基底関数を第 p 動径基底関数として複製する．そのとき，シナプス結合荷重 w_p，パラメータ Σ_p 並びにパラメータ m_p は形質としてもとの第 j 動径基底関数のものを引き継ぎ，パラメータ m_p は新たに増えた点とする．STEP 1. へ戻る．

第3章

機械学習に関して

§ 3.1　機械学習と最適化の関係

現在，機械学習は学術界だけでなく産業界においても幅広く用いられ，人工知能技術のコア技術として重要な役割をはたしている．機械学習はもともと人と同様の知能的機能を実現させるために研究開発が進められてきた分野であるが,「データから学ぶ」という過程とデータ解析・統計学の方法論がうまくマッチし，現在では狭い意味での人工知能としての使い方にとどまらず幅広いデータ科学の方法論として発展している．初回の今回は機械学習の概要として機械学習のおおまかな歴史とその代表的な問題設定および機械学習を実行するためのツールについて述べる．

教師あり学習

教師あり学習は機械学習の中でも特に基本的な問題である．教師あり学習において目指すべき目標点は，ある入力 $x \in X$(画像やテキストなど) に対して，そのラベル $y \in Y$(画像に写っている物体や人など) を予測することにある．訓練データとして n 個の入出力データの組 $(z_i)_{i=1}^n = (x_i, y_i)_{i=1}^n$ が得られているとして，X から Y への関数 $h(x)$ を考える．損失関数は正解ラベル y と $h(x)$ の差を評価する関数として 2 乗誤差評価関数 $E(y, h(x))$ が用いられることが多い．

$$E(y, h(x)) = \frac{1}{n}\sum_{i=1}^{n}(y_i - h(x_i))^2 \tag{3.1}$$

教師あり学習の基本的な問題として回帰がある．回帰問題においては $Y = \mathbb{R}$ で $X = \mathbb{R}^p$ のときに $h(x)$ として線形関数 $h(x) = a^{\mathrm{T}}x$(ただし，$a \in \mathbb{R}^p$) を用いれば線形回帰になる．このときの訓練誤差の最小化

$$\min_{a\in\mathbb{R}^p} \frac{1}{n}\sum_{i=1}^{n}(y_i - x_i^{\mathrm{T}}a)^2 \tag{3.2}$$

は最小二乗法と呼ばれている．

教師なし学習

教師なし学習は教師あり学習と違い，入力に対するラベルが付いていない．このような問題は，分類に代表されるクラスタリングや，データの低次元への

圧縮として考えられる．クラスタリングの場合，観測データからその裏にある真の分布を推定することで実現されることが多い．例えば，混合ガウス分布の密度関数をデータにあてはめることでソフトクラスタリングが実現できる．

強化学習

強化学習は環境に合わせて自ら最適な行動を学ぶ学習問題である．例えばゲームを解く AI やロボットの動作学習などに用いられる．実際に，AlphaGo は強化学習を要素技術として用いている [5]．強化学習は能動的にデータを取得しながら学習するという点で，上記の学習方法とは異なっている．その応用範囲の広さから深層学習による応用研究が盛んに行われている．強化学習の基本的な手法である価値反復法について説明する．強化学習では状態 $s \in S$(ゲームの局面など) と行動 $a \in \mathbb{A}$(その局面に対して行う行動) が基本要素になる．$P(s'|s,a)$ を状態 s で行動 a を取ったときに，次の時刻に状態 s' に移る遷移確率を表し，$R(s)$ で状態 s に到達したときに得られる報酬を表す．例えば迷路を解く問題であれば，$R(s)$ としてゴールすることで高い報酬が得られる関数に設定する．ある方策 π(各状態で次に起こす行動を決める関数) に従って行動するときの報酬の総和の期待値を

$$U^{\pi}(s) = E[\sum_{t=0}^{\infty} \gamma^t R(s_t)|\pi, s_0 = s] \tag{3.3}$$

と定義する．これは，状態 s から初めて未来に得られる報酬の総和であり，$\gamma < 1$ は時間割引を表す係数である．強化学習が目標とするのは報酬 $U^{\pi}(s)$ を最大にする方策 π を求めることにある．もし，$U^{\pi}(s)$ が分かっているのなら，最適な方策は期待報酬を最大化する行動を取るので，

$$\pi^*(s) = \underset{a \in \mathbb{A}}{\mathrm{argmax}} \Sigma_{s' \in S} P(s'|s,a) U^{\pi^*}(s) \tag{3.4}$$

を満たすはずである．このとき，U^{π} の定義から

$$U^{\pi^*}(s) = R(s) + \gamma \underset{a \in \mathbb{A}}{\mathrm{max}} \sum_{s' \in S} P(s'|s,a) U^{\pi^*}(s') \tag{3.5}$$

が満たされる．これを Bellman 方程式という．しかし，実際は π^* も $U^{\pi^*}(s)$ も分からないので，状態空間を遷移しながら Bellman 方程式に従って各遷移ごとに $U^{\pi}(s)$ を更新し収束するまで続ける．このような方法を価値反復法と呼ぶ．価値反復法は $P(s'|s,a)$ を知っている必要があるが，実際はこれも推定する必要がある．これも考慮した学習方法が Q-学習である．また，深層学習を用いた Q-学習の方式として Deep Q-network が提案されている．

§ 3.2　エネルギー関数の最小化手法

機械学習はポテンシャルやエネルギー関数を最小化するための探索手法によって種類分けされる場合がある．

最急降下法

最急降下法は関数の傾き（一階微分）のみから，関数の最小値を探索する連続最適化問題の勾配法のアルゴリズムである．最急降下法では目的関数として与えられる $f(\mathbf{x})$ を，n 次のベクトル $\mathbf{x} = (x_1, x_2, ..., x_n)$ を引数とする関数として，この関数の極小値を求めることを考える．勾配法では反復法を用いて $\mathbf{x}$ を解に近づけていく．k 回目の反復で解が $\mathbf{x}^k$ の位置にあるとき，最急降下法では次のようにして値を更新する．

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} - \alpha \mathrm{grad} f(\mathbf{x}^{(k)}) \tag{3.6}$$

ここで α は 1 回に更新する数値の重みを決めるパラメータであり，通常は小さな正の定数である．この値が大きすぎると解が発散する可能性があり，小さすぎると収束が遅くなる可能性がある．適切な α を選ぶことは，アルゴリズムの性能に大きな影響を与える．

焼きなまし法

焼きなまし法は大域的最適化問題への汎用の乱択アルゴリズムであり，広大な探索空間内の与えられた関数の大域的最適解に対して，よい近似を与える．焼きなましとは金属工学の用語であり，金属材料を熱した後で徐々に冷やし，結晶を成長させてその欠陥を減らす作業である．焼きなまし法では，探索空間の各点「s」は物理システムの「状態」に対応し，最小化すべき関数 $E(s)$ は物理状態の「内部エネルギー」に対応する．したがって，目標はシステムを任意の「初期状態」からできる限りエネルギーが最小の状態にすることである．

1. 基本的反復
   各ステップでは、SA のヒューリスティックは、現在状態 s のいくつかの近傍 s' を検討し、現在状態 s のままがよいか、いずれかの近傍状態に遷移するのがよいかを確率的に決定する。その際、システムが最終的にエネルギーの低い状態へ向かうよう考慮する。このステップは、十分よい結果が得られるまで、あるいは予定された計算時間が尽きるまで繰り返される。
2. 遷移確率
   現在状態 s から新たな状態候補 s' への遷移確率は、二つの状態のエネルギー e = E(s) と e' = E(s') の関数 P(e, e' , T) で与えられる。ここで T は「温度」に相当し、時間と共に変化するグローバルなパラメータである。
   遷移確率 P の基本的な必須条件として、e' ¿ e のときにゼロでない値を返さなければならないという点があげられる。これは、ときには「悪い」と思われる状態（エネルギーの高い状態）へもシステムが遷移可能であることを意味する。これは「ローカルな極小状態」に張り付いてしまうのを防ぐ機能である。「ローカルな極小状態」とは、そのエネルギーが真の極小には程遠いが、近傍とだけ比べれば極小であるような状態を意味する。
   一方で、T が 0 に近くなるにつれて、e' ¿ e であれば P(e, e' , T) の値をゼロに近づけ、e' ¡ e であればその値を大きくする。これにより、T が十分に小さくなれば、システムは極小に向かい、逆の動きは封じられる。特に T が 0 になると、山登り法を使うことで近傍の極小に確実に向かわせることもできる。

関数 P は e’ − e の値が増大する際には確率を減らす値を返すように設定される。つまり、ちょっとしたエネルギー上昇の向こうに極小がある可能性の方が、どんどん上昇している場合よりも高いという考え方である。しかし、この条件は必ずしも必須ではなく、上記の必須条件が満たされていればよい。

これらの特性により、状態 s の変化は温度 T に大きく依存する。大まかに言えば、s の変化は T が大きいときには劇的に変化し、T が小さくなるとゆるやかに変化する。

3. 焼きなましスケジュール焼きなまし法のもうひとつの本質的な機能は、シミュレーションが進むにつれて、温度が徐々に下がっていく点である。最初 T は高い（あるいは無限大の）値に設定され、何らかの「焼きなましスケジュール」に従って、ステップを経るにつれて減少していく。そのスケジュールは、ユーザーが指定する場合もあるが、予定された時間には T =0 になって終わらなければならない。このようにすると、システムは最初のうちはエネルギー関数の小さな変化を無視して最適解を求めて探索空間の広い領域をさまよい、徐々にエネルギーの低い領域に向かって探索範囲を狭めていき、最終的に最急降下法のヒューリスティックに従って最もエネルギーの低い状態に降りていくのである。

§ 3.3　最小化する方法

第4章

提案手法

§ 4.1 Google Patentsからのデータ収集の高速化と分類

本研究では，Google Patentsから特許情報をスクレイピングすることで収集する．まず特許番号を取得し，それらの番号を用いて特許が表示されているページにアクセスし，そこから特許本文をテキストとして取得する．ユーザーが指定したキーワードのor検索をGoogle Patentsで行う．Google Patentsにおいてor検索を行うにはワードとワードの間にスペースを開ける必要がある．Google Patentsでは一度に1000件までしか表示することができない．そのため，それぞれが1000件を超えないように年代を1年ごと区切ってスクレピングを行う．特許の出願日がその年の1月1日から12月31日である特許を取得した．取得したデータを図4.1に示す．

本研究の提案手法は，大別すると以下のような工程からなる．

1. 利用者の入力したワードにおけるGooglePatentsでの検索結果を取得する．
2. 取得した特許データの本文に対してSentence-BERTを用いてベクトル化を行う．
3. 出力されたベクトルに対して次元圧縮を行う．
4. 次元圧縮を行ったデータに対してシルエット分析を行いクラスタリングする．
5. それぞれのクラスターについて，K-meansで求めた重心からのユークリッド距離の近い10個のデータを用いて各クラスターのタイトルを作成する．
6. ユーザーが指定したクラスターに対して，Simpson係数を用いて共起関係を導出する．
7. 求めたSimpson係数をもとに3Dグラフおよび2Dグラフを作成する．

システムのフロントページおよび対象の選択

提案手法においてユーザサイドに提示されるフロントページを図4.2に示す．図4.2に示した通り，システムのはじめにユーザにはキーワードの入力画面が表示される．キーワードの入力については一つの単語であればそのまま入力し，複数単語入力したい場合は単語と単語との間にスペースを空けて入力することで入力することができる．また，取得するデータの年数を指定することができる．初期設定では直近6年間のデータを収集するようになっているが，キーワードの内容によっては6年では十分な量のデータを取得できない場合があるため，ユーザー側で取得する年数を指定できるようにしている．このページにおいてユーザは自身が対象としたいキーワードおよび取得するデータの年数を指定する．

ここで，スクレピングを行う際に時間がかかってしまう問題を解決するために，pythonのモジュールthreadsを用いてマルチスレッドによるスクレピングを行う．コンピュータの

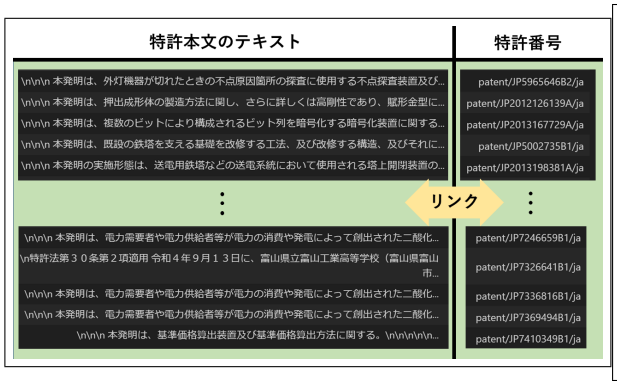


図 4.1: テキストデータのフォーマット

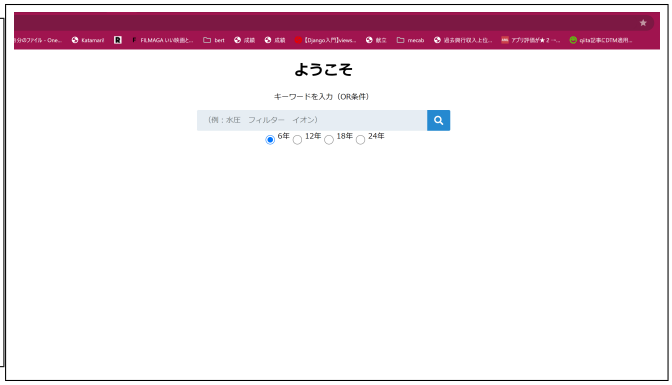


図 4.2: システムのフロントページ

性能によって並列にする数を増やすとかかる時間が長くなる．そのため，並列にする数を 6 つにして実行を行った．

threads

スレッドベースの並列タスク実行を助けるモジュールである．threads を利用することで，一つの python プログラムの中で複数の処理を同時に実行するマルチスレッド処理を実現できる．threads モジュールには Threads クラスが定義されており，この Threads をスーパークラスとして新しいスレッドを定義する．run（）メソッドの中にそのスレッドが実行する処理を書き，start（）メソッドを呼び出すことでスレッドが起動し，並列で処理が進む．

join（）メソッドを使用すれば，スレッドの終了を待って制御することもできる．Lock や Semaphore，Event などの動機機構も利用でき，スレッド間でデータを安全に共有することも可能である．ファイル IO やネットワークアクセス時の待ち時間を隠蔽したり，応答性を向上させたり，並列処理による速度向上が図れるなど，threads は即効性の高い並列プログラミングを実現できる．

提案手法では，対象期間を 1 年ごとに分割し，それぞれの期間を個別のスレッドに割り当てて並列処理を行う．具体的には，6 つのスレッドを作成し，各スレッドが 1 年間のデータをスクレイピングする．つまりスレッド 1 は 1 年分，スレッド 2 は次の 1 年分となる．そして，各スレッド内で 1 年ごとにデータ収集を行う．こうすることで期間ごとの並列処理が可能となり，スクレピングの速度や効率を向上させることができる．最後に，すべてのスレッドが実行完了後，収集したデータを統合することで，対象期間全体のデータセットを取得する．

データの分類

収集したテキストデータをもとに Sentence-BERT を用いてそれぞれをベクトルに表現する．この時，Sentence-BERT によって出力されるベクトルを pickle を用いて保存しておく．それらのベクトルを UMAP を用いて 15 次元および 2 次元のベクトルに圧縮する．この際，UMAP に設定するパラメータについて説明する [?].

パラメータの設定

n_neighbors

n_neighbors パラメータは，各データポイントの埋め込みにおいて考量する近隣点の数を指定する．この値が大きいほどデータ全体の構造が強調され，小さいほど局所的な構造が強調される．小さな n_neighbors 値（5-20）は，小規模なクラスタや微細な構造の検出に適していおる．一方大きな n_neighbors 値（50-200）は，データ全体の構造や大規模なクラスタを強調するときに用いられる．

min_dist

min_dist パラメータは，UMAP によって生成される低次元埋め込み空間内のデータ点間の最小距離を制御する．小さな min_dist 値（0. 0-0. 3）はデータが密集したクラスタリングを得る際に適用する．中程度の min_dist 値（0. 3-0. 7）は，クラスタ間のバランスが取れた埋め込みを得る際に適用する．大きな min_dist 値（0 . 7 以上）は，クラスタが広がり，隣接するクラスターとの距離を最大化する場合に適用する．min_dist パラメータは低次元空間内のデータ点は位置をコントロールし，クラスタの密度やスペースを調整する役割がある．

n_components

n_components パラメータは，UMAP によって生成される埋め込み次元の次元数を指定する．n_components=2 や n_components=3 の低次元埋め込みは，データの分布を直観的に把握しやすいため，結果の可視化を目的とする場合に選択される．一方で n_components が 3 以上の値を設定した場合，特定のアルゴリズムでの利用や，次元削減後のデータをほかの分析タスクに利用されることを目的とする．n_components パラメータの値は解析目的やデータ利用法に応じて適切に定める必要がある．

metric

metric パラメータは，データ間の類似度や距離を算出するための手法を指定することができる．これにより，データ空間の幾何学的性質が定義される．数値データの場合，ユークリッド距離やマンハッタン距離などの標準的な手法を指定するのが一般的である．一方テキストデータの場合には，コサイン距離やハミング距離などのテキスト向けの手法が利用される．データの型や構造に応じた適切な手法を metric パラメータに設定することで，UMAP のパフォーマンスが最大化される．

15 次元のベクトルはクラスタリングを行う際と，クラスターの解釈を行う際に用いる．2 次元のベクトルはクラスタリングを行ったデータをプロットする際に用いる．プロットする際に 2 次元ベクトルを用いる理由は，3 次元ベクトルやそれ以上の次元数のベクトルと比較して，2 次元のベクトル空間上にプロットされた各データ点間の距離感や密集具合を人間の知覚として把握しやすいためである．また，データ間の類似性を可視化する上でも，2 次元空間上では各クラスター内でのデータ点のまとまり方を把握しやすく，データセット全体の構造を俯瞰しやすい．

§ 4.2　クラスターの解釈と共起語ネットワーク

クラスターの解釈を行うために各クラスターの重要語を表示する．ここで，各クラスターのすべての点を対象に重要語を計算しようとすると，データの数によっては，莫大な処理時間になる可能性がある．そのため，計算コストを抑えつつ各クラスターの特徴を表すデータを効率的に取得する必要がある．

そこでまず，各クラスター内で最も代表制の高いデータを簡易的に抽出することを試みる．具体的には，K-menas アルゴリズムによって求められた各クラスターごとの重心に最も近いデータをユークリッド距離を利用して近い順にソートし上位から 10 個ずつ取得する．これにより各クラスターの典型的な特徴を示すと考えられるデータを効率よく抽出できる．

その後，この抽出したデータに対して各クラスターごとに重要語や特徴語を計算する．これにより各クラスターの概要や内容の傾向を効率かつ低コストで把握することができる．このような処理フローを設定することで，大規模データに対しても実現できな時間でクラスターの解釈や分析を行うことができる．

ユークリッド距離

ユークリッド距離は，座標空間において 2 点間の直線距離を表す指標である．2 点をそれぞれ $(x_1, y_1, z_1, \dots)$ および $(x_2, y_2, z_2, \dots)$ としたときの座標間のユークリッド距離は以下の式のように求められる．

<n 次元空間の場合 >

$$d = \sqrt{\sum_{i=1}^{n} (x_{2i} - x_{1i})^2} \tag{4.1}$$

得られたデータに対しては，専門用語や複合語を考慮した重要語の計算を行う．termextract を用いて，データ内に含まれる重要なキーワードや専門用語を抽出する．最終的に，各クラスターいおいて重要度が高い単語を 3 つ選出し，それを解釈として表示する．これによってクラスターが表す内容やグループの性質を的確に把握することができる．

システムのグラフ表示およびクラスターの選択

実際に作成された散布図を画像データとして保存する．散布図の生成には pyvis を用いる．その際画像の中のクラスタはそれぞれ別の色でプロットして，それらのクラスターの番号を補足情報として画像に追加する．またそれぞれの点がでかすぎて画像が見づらくなることを加味し，点のサイズをあらかじめ設定しておく．さらに，クラスターの数が多くなると色の違いによるクラスターの判別が難しくなるため，それぞれのクラスタウーの点の形を変えることで色だけでなく形でもクラスターを区別できるようにしている．そのあと画像データを html 上に表示する．また，各クラスターの内容とそれらのクラスター番号ををそれぞれ箇条書きで表示する．画像のクラスターとその内容を照合することで，ユーザーが任意のクラスターを指定できるようにする実際の解釈の出力例を図 4.3 に示す．

データの前処理

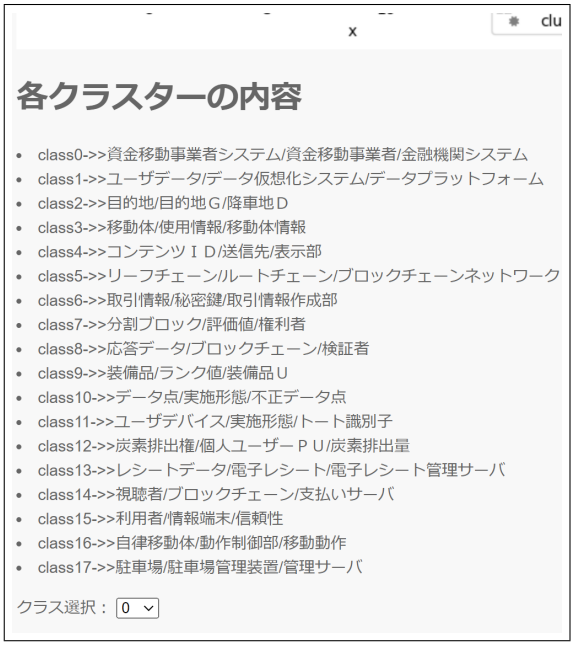
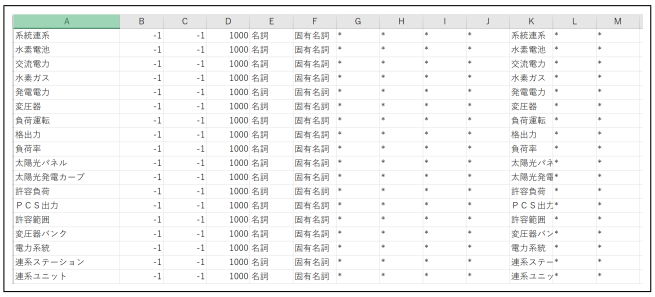


図 4.3: 解釈の出力例



A	B	C	D	E	F	G	H	I	J	K	L	M
系統連系	-1	-1	1000	名詞	固有名詞	*	*	*	*	系統連系	*	*
水素電池	-1	-1	1000	名詞	固有名詞	*	*	*	*	水素電池	*	*
交流電力	-1	-1	1000	名詞	固有名詞	*	*	*	*	交流電力	*	*
水素ガス	-1	-1	1000	名詞	固有名詞	*	*	*	*	水素ガス	*	*
発電電力	-1	-1	1000	名詞	固有名詞	*	*	*	*	発電電力	*	*
変圧器	-1	-1	1000	名詞	固有名詞	*	*	*	*	変圧器	*	*
負荷運転	-1	-1	1000	名詞	固有名詞	*	*	*	*	負荷運転	*	*
格出力	-1	-1	1000	名詞	固有名詞	*	*	*	*	格出力	*	*
負荷率	-1	-1	1000	名詞	固有名詞	*	*	*	*	負荷率	*	*
太陽光パネル	-1	-1	1000	名詞	固有名詞	*	*	*	*	太陽光パネ	*	*
太陽光発電カーブ	-1	-1	1000	名詞	固有名詞	*	*	*	*	太陽光発電	*	*
許容負荷	-1	-1	1000	名詞	固有名詞	*	*	*	*	許容負荷	*	*
ＰＣＳ出力	-1	-1	1000	名詞	固有名詞	*	*	*	*	ＰＣＳ出力	*	*
許容範囲	-1	-1	1000	名詞	固有名詞	*	*	*	*	許容範囲	*	*
変圧器バンク	-1	-1	1000	名詞	固有名詞	*	*	*	*	変圧器バン	*	*
電力系統	-1	-1	1000	名詞	固有名詞	*	*	*	*	電力系統	*	*
連系ステーション	-1	-1	1000	名詞	固有名詞	*	*	*	*	連系ステー	*	*
連系ユニット	-1	-1	1000	名詞	固有名詞	*	*	*	*	連系ユニッ	*	*

図 4.4: ユーザー辞書のフォーマット（csv）

共起語ネットワークを作成する際に，文章を分かち書きする必要がある．この時，特許には多数の専門用語や複合語が含まれるため，それらを抽出したうえで分かち書きを行う．termextract では専門用語の抽出を行うことはできるが，それらを用いて分かち書きを行うことはできない．そこで，今回用いた分かち書きのモジュールである Janome にユーザー辞書として専門用語や複合語を登録する．

ユーザー辞書のフォーマット

Janome は独自の単語や品詞情報を追加することができる．この機能を用いることで，特定の文脈や専門用語に適した分かち書きを行うことができる．csv 形式でユーザー辞書を与える必要がある．

＜ユーザー辞書の形式＞

> ユーザー辞書はカンマ区切り CSV ファイルで,「表層形，左文脈 ID，右文脈 ID，コスト，品詞，品詞細分類 1，品詞細分類 2，品詞細分類 3，活用型，活用形，原型，音読み，発音」という形式で与える必要がある．今回は表層系に抽出された専門用語や複合語を入力し，左文脈 ID および右文脈 ID は-1 に指定する．コストはすべて 1000 とし，品詞には名詞，品詞細分類 1 には固有名詞を設定する．品詞細分類 2，品詞細分類 3，活用形，活用型，音読み，発音はは未設定とし原型においては表層系と同じ文字列を入力する．実際に作成された CSV ファイルを図 4.4 に示す．

この時，作成される辞書の量が多くなると Janome の分かち書きが正しく動作しないことがある．それらを解決するために辞書の量をあらかじめ削減する．削減する方法は事前に求めた専門用語や複合語の重要度が低いものを優先的に削除していく．

分かち書きを行ったのち，共起語ネットワークを作成する．クラスターごとに共起語を分析する．共起語を分析する際，一般的な用語が多く含まれてしまうことがあるため，重要

度が高い単語が優先的に含まれるようにする．事前に計算した単語の重要度を用いて，共起語の中に重要ではない単語が含まれているものを除外する．

共起語ネットワーク

本研究では，単語間の共起関係を分析するために Simpson 係数を用いる．Simpson 係数は Jaccard 係数や Dice 係数と比較して，差集合の要素数による影響をより小さく抑えることができる．ただし，Simpson 係数は一方の集合が他方の真部分集合である場合に 1 となる．そこで，今回，Simpson 係数が 1 となる場合には，それらがもともと一つの単語であったとみなして分析対象から除外している．また，片方の集合の要素数が極端に少ないと係数値が大きくなる傾向があることから，お互いの集合数に 5000 以上の開きがある場合も分析対象から除いている．さらに，共起関係を求める際に，一般的な用語が多く出現する傾向があるため上記で求めた，単語の重要度を用いて重要度が高いものを優先に分析を行う．上記の分析を用いて作成された共起語ネットワークを 3D グラフおよび，2D グラフによって可視化を行う．

Three.js を用いた 3D グラフ作成

3D グラフの作成には Three.js のモジュールである 3D Force-Directed Graph を用いる．3D Force-Directed Graph グラフでは Json ファイルの形式でデータを与えることができる．先ほど作成された共起語ネットワークを，ノードの名前と，矢印の元のノードおよび矢印の先のノードを ”nodes ”および ”links ”として与えることでグラフの描画に必要な情報を受け渡す．受け渡された情報をもとにグラフを作成する．ここで，3D Force-Directed Graph の初期設定ではグラフのノードは球体になっており，テキストの表示を行うには適しているとは言えない．そこでノードそのもをテキストにする．さらに，3D グラフにおけるノード間の線には共起元の単語から共起先の単語への矢印を描画しているが，ノード間の距離が広いと矢印による識別が難しくなる．そこで矢印の方向方向に向かって流動的なアニメーションを追加している．他にも，読み込まれたグラフのカメラ操作はグラフの中心を軸に 360 度回転することができるが，ノードの場所によってはあまり詳細に表示できない場合がある．そのためノードをクリックすることでそのノードを中心とする回転に変更でき，そのノードを中心としてノードの付近を見渡すことができる．

pyvis を用いた 2D グラフ作成

2D グラフの作成には pyvis を用いる．pyvis では共起元のワードと共起先のワードをノードとして追加し，それらの関係について定義することでグラフを描画することができる．また，初期設定では，ノード間の線は矢印にはなっていない．そこでグラフを描画する際の設定 directed という項目の設定を変更する必要がある．また，2D グラフでは，共起関係の強さによって矢印の太さを変更している．共起関係が強いものは太く，弱いものは細くなるようにしている．このことで，一目で単語同士の共起関係の強さを把握することができる．さらに，ノードをクリックすることで，そのノードに向かっているノードとそのノードから向いているノードを強調表示することができる．

§ 4.3　システム化とIPランドスケープへの活用

4章で示した各手法を統合した課題解決のための提案手法全体の流れの説明を行う．また，これまでに説明した技術のそれぞれがどの部分に組み込まれているかについて整理しながら，flaskを用い作成した提案手法を組み込んだシステム全体の流れを説明する．提案システム全体のフロー図を図4.5に示す．

flaskアプリケーションの作成

ユーザーからの入力や，結果の出力を行うためのアプリケーションをflaskを用いて作成した．flaskとはpythonでWebアプリケーションを作成するための軽量なフレームワークであり，flaskの最大の特徴は軽量性とシンプルさである．設定やコードが少なく，簡単にアプリケーションの開発を行うことができる．加えて，flaskは拡張性に優れている．必要に応じて様々なライブラリを使用することで，機能を拡張することができる．また，URLの経路設定や経路処理，テンプレート，データベースなど重要な機能を柔軟に組み合わせることができるため，目的に合ったアプリケーションを作成することができる．

提案手法全体の流れ

Step 1: キーワードの入力・取得年数の選択

フロントページにてユーザーからのキーワードの入力を取得する．一つの単語だけでなく複数の単語でも検索できるようにすることで広い範囲の検索を可能にする．複数の単語を入力したい場合は単語と単語の間にスペースを空けて入力することで，複数の単語の入力を行うことができる．またここで取得したい年数を指定することができる．初期設定は6年間となっており，2017年から2023年までのデータを取得することができる．それ以外にも12年，18年，24年と選択することができる．

このようにユーザーが選択できるようにすることで，データの取得数が不足，または，過剰となることを回避することができる．このことにより，結果の質的向上を目指いしている．また取得するデーターが多いと，スクレイピングに時間がかかってしまうため，それらの回避も行うことができる．ユーザーの用途や目的によって年数を選択することができる．

ユーザーが入力した情報をもとにリアルタイムでスクレピングを行う．ただし，検索を行ってからスクレイピングを行っている間，画面がそのままであると待機状態が不明瞭となるため，スクレイピングを行っている際は，処理時間専用の画面を用意している．入力されたキーワードをもとにスクレイピングを行ったデータは次のフローへ送信される．

Step 2: 特許の俯瞰とクラスターの選択

Step1で取得したデーターをもとにしたクラスタリングの結果をプロットする．これらのプロットの結果を見ることで特許全体を俯瞰することができる．また，それらのなかでの分野のまとまりについても見ることができる．各クラスターについてはそ

れぞれ違う色の点で描画しており，それらのまとまりをより視覚的にわかりやすいようにしている．ここで，色だけの差別化では，色による違いが判らない場合があるため，それぞれの点の形を変更することで，それぞれのクラスターの区別を行っている．

さらに，クラスターの解釈を図の中に組み込んでしまうと図と文字が重なって見づらくなるため，クラスターの解釈については図の外に記述してある．ユーザーは図の中クラスターの番号と解釈におけるクラスターの番号を照らし合わせることでそれぞれのクラスターの解釈を確認することができる．クラスターの解釈をもとにユーザーは自身の興味のあるクラスターを選択することができる．クラスターの選択は自身の選択したクラスターの番号を選択し，送信ボタンを押すことで，システムにその情報が送信される．

以上のことにより，ユーザーは自分が知りたい技術分野や，特許の散らばり具合から，密になっている部分や疎になっている部分に対して分析の対象を選択することができる．例えば特定分野が集中的にクラスタになっている部分や，逆に分散している部分からそれらのクラスターを分析対象とすることができる．さらに，散布図での出力により，特許全体やそれぞれのクラスターからなる技術のトレンドなどを一目で把握することができると考える．

Step 3: 選択されたクラスターにおける共起語ネットワークの作成

Step2 にて選択されたクラスターにおける共起語ネットワークを作成する．Simpson 係数を用いて共起関係の分析を行う．そこで計算された係数値をもとに 3D グラフおよび 2D グラフによる可視化を行う．ここで，グラフを表示するときのグラフの大きさを設定できるようにしている．ユーザーによって分析の目的やニーズによって異なるケースが考えられる．広い範囲を分析することで広域的な分析を行いたい人，また狭い範囲で狭域的な分析を行いたい人など，様々なケースが考えられるので，出力されるグラフの大きさをユーザーが指定することができるようにしている．このことにより，ユーザーは自分の分析目的に合わせて適切なグラフの大きさを設定することが可能であり，広い範囲を見渡すことで全体像を把握したり，狭い範囲で詳細な関係性を分析したりすることが可能である．

また，選択されたクラスターに含まれている特許の特許番号を一覧に表示する．特許をスクレピングする際に取得された特許番号と各ベクトルが紐ずけられているためクラスターの中に含まれるベクトル情報からクラスターに含まれる特許番号を取得する．さらにその特許番号をクリックすることで，元の特許における GooglePatents でのページが表示されるようになっている．このことで，実際の特許にもアクセスすることが可能となる．

以上の操作の結果もとめられた共起元の単語と共起先の単語およびそれらの Simpson 係数をデータフレームに保存し pickle データの形式で保存する．ここで，pickle を用いるのは，csv 形式で保存してしまうと，

Step 4: 共起語ネットワークの可視化

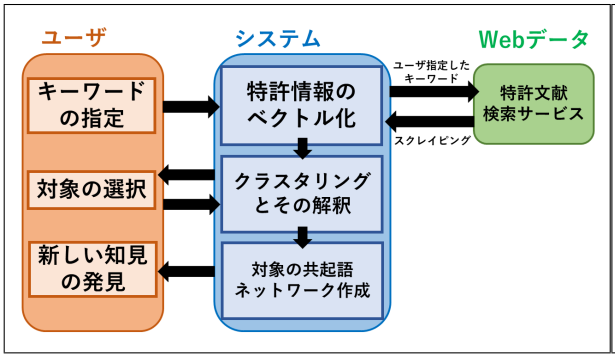


図 4.5: 提案システム

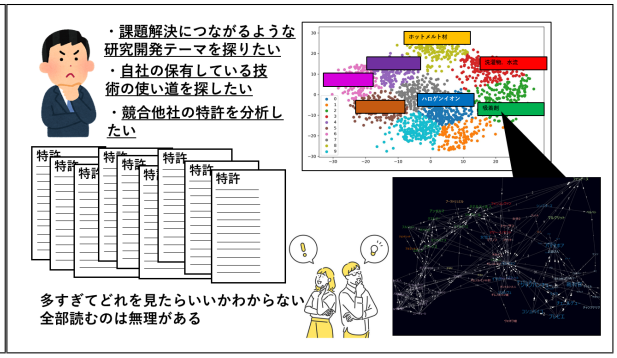


図 4.6: IPL への活用

Step3 で作成された共起元の単語と共起先の単語および Simpson 係数の値を用いて言共起語ネットワークを 2D および 3D グラフによって可視化する．グラフの作成には Three.js のモジュールである 3D Force-Directed Graph を用いる．

Json ファイルで共起元の単語と共起先の単語を json ファイル形式で与える．与えたファイルをもとに 3D グラフを描画する．描画されたグラフはマウスをドラッグすることで回転でき，異なる視点からの観察が可能である．またホイールを回転させることでグラフの拡大，縮小を行うことができる．さらに，単語をクリックすることでその単語を中心とした回転に変更することができる．加えてグラフの矢印の向きは線の途中に描画してあるが見づらい部分があるため，流動的なアニメーションを追加することでわかりやすくしてある．上記で説明したシステムの実際の IPL への活用例を図 4.6 に示す．

第5章

実験結果並びに考察

§ 5.1　実験の概要

本研究における提案手法においてIPランドスケープ実施への支援が行えているかに注目して評価実験を行う．IPランドスケープの取り組みとして，技術の特徴を生かした有望用途の探索を行うことを目的とする．今回の評価実験では，IPランドスケープの一環として特許情報の探索およびその中から知見を発見することを目的として検証を行う．実際には，自社の保有している技術の使い道を探すという題材をもとに検証を行う．

そのため,「ブロックチェーン技術を活用した決済システムの特許分析」という事例を設けて実験を行う．実際にシステムの入力欄に「ブロックチェーン」「決済システム」という単語を検索欄に入れ検索年数を6年にして実行を行った．

UMAPに設定するパラメータはn_neighborsの値はあまり大きなクラスターにしてしまうとそれぞれの要素の数が多くなってしまい大まかな分類になってしまうことを踏まえ「25」に設定し，min_distの値は出力されるクラスターの密度やスペースの具合を加味し「0.1」，metricは今回用いるデータがテキストを定量化したデータであるため「cosine」に設定して実験を行った．また，3Dグラフを描画するときに指定できる大きさの設定は表示する共起関係の数であり，小は1000，中は2000，大は3000個の共起関係を表示している．

さらに，実際にシステムを使用してもらい，アンケートに答えてもらう．アンケートの項目は全部で10個あり，その10個には必ず答えてもらう．以上の項目を5段階評価のリッカート尺度による評価を行ってもらう．今回のアンケートでは5段階評価のうち，1を「まったく満足していない」，2を「あまり満足していない」，3を「どちらでもない」，4を「やや満足している」，5を「非常に満足している」といったようなアンケートを行った．

また実際のシステム利用時の大剣を直接フィードバックできるように，アンケートと同時にコメントを入力できる欄を設けて置き，実際に入力したキーワードなどを自由にコメントができるようにする．実際のアンケート内容を表5.1に示す．表5.1を見てわかるように，アンケートの半分についてはシステムの使用感についての質問を設定している．システムの使用感についての質問から客観的なシステムの使用感に関する質問を行っている．残りの半分はそれぞれの機能についての質問を行っている．システムから出力されたものが適切であるかに関する質問を行う．

調査の対象は同研究室の学部4年生，3年生の合計5人に実際に開発したシステムを使用してもらい，アンケートを答えてもらった．実験では，利用者に実際にキーワードを考えてもらい，それを検索欄に入力することを行った．また，取得する年数に関しては，まずは6年を指定してもらい，得られた結果が少ない場合は徐々に年数を上げていくというこ

表 5.1: アンケート内容

システムの操作性はわかりやすいか	システムの機能は理解しやすいか
レイアウトは親切か	デザインは見やすいか
ストレスなく利用することができたか	クラスターの内容を理解することができたか
共起語ネットワークによる出力は適していたか	3Dグラフによる提示は適切であるか
効率的な特許探索を行えると思ったか	新しい知見を発見できそうか

とを行った．

この評価を通じて，本手法が IP ランドスケープの支援に役立つ実践的支援機能を果たしているかどうかの確認を行う．

§ 5.2　実験結果と考察

まず，事例を設けての実験についての結果と考察を行う．この時 1588 個の特許をスクレイピングすることができた．実際に出力された散布図は図 5.1 となり，クラスターは 17 個となった．それぞれのクラスターに対応したタイトルは図 5.2 のような出力となった．

クラスターにおいて 3D グラフからブロックチェーンの使い道を検討した．クラスター 4 を選択した際に出力された 3D グラフを図 5.3 に示す．出力された 3D グラフから「コンサート」や「グッズ」などから「ファン通貨」という単語につながりがあるこが把握的た．この関係性から，アーティストのファン特有の通貨をブロックチェーン技術を用いて作り出し，ファンのコミュニティー内でその通貨を発行することが考えられる．通貨を獲得するには，アーティストのコンサートなどに行ったり，それらの情報を外部に発信したときなどがあげられる．ファンがこれらの活動を行うことで，通貨を獲得することができる．この通貨を用いることでファンコミュニティ独自の決済システムを採用することが可能となる．

ファンはこの通貨を使用してアーティスのグッズやコンサートチケットなどを購入することができる．また，通貨の利用により，ファン同士の交流やコミュニティの活性化を促進することも期待することができる．このようなファンコミュニティ独自の通貨システムは，アーティストとファンの絆を深めるだけでなく，ファンの忠誠心や参加意欲を高める効果も期待できる．さらに，ブロックチェーンを用いることで，通貨の取引履歴や所持数などの透明性や信頼性を確保することもできる．

したがって，出力されたグラフの結果から，アーティスト特有の通貨を作り出し，ファンコミュニティ内で利用することで，独自の決済システムを実現するということを考えることができる．

最後にアンケート調査における結果と考察を行う．

一個目に,「システムの操作性はわかりやすいか」という質問を行った．結果として，全体的に好印象な評価を得ることができた．この結果から．システムの操作性は容易であることが考えられる．システム全体的に直観的に操作できるということが考えられる．

二個目に,「システムの機能は理解しやすいか」という質問を行った．結果として，好印象な評価が四人であったが，残りの一人に関してはどちらでもないという意見であった．こ

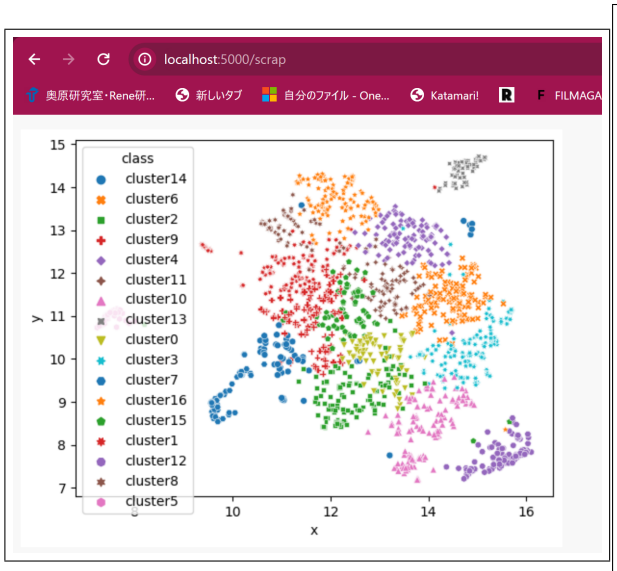


図 5.1: ベクトル化の結果

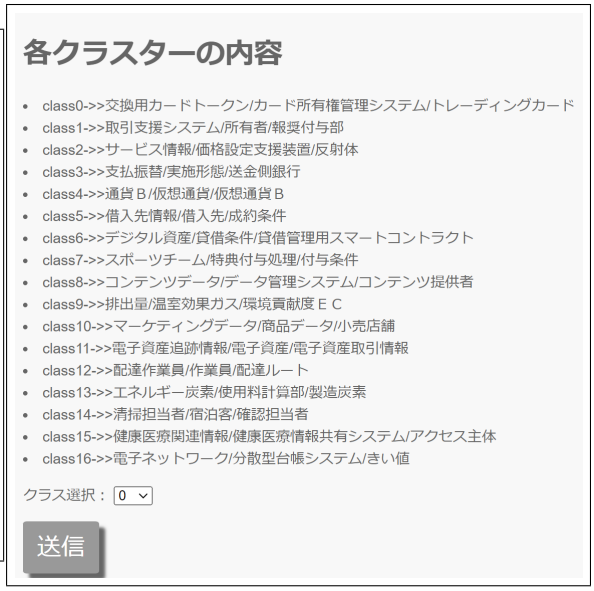


図 5.2: 出力されたタイトル

の結果からシステムを初めて使う人でもある程度すぐにシステムの機能を理解することができるということがわかる．また，もう少し画面に出力されているものがどういうものなのかを説明することで，よりシステムの機能を理解してもらうことができると考えられる．

三個目に,「レイアウトは適切か」という質問を行った．結果として，全体的に好印象な評価を得ることができた．この結果から．グラフやボタン，テキストなどの表示位置が適切であると考えることができる．

四個目に,「デザインは見やすいか」という質問を行った．結果として，全体的に好印象な評価を得ることができた．この結果から，本システムの画面全体を通してのデザインが見やすいということが考えられる．画面に表示する情報は必要最低限にしているためであると考えることができる．一方で，二個目の質問で考察したように，システム機能の説明を付け加えることを考えると，デザインの構成を考える必要がある．

五個目に,「ストレスなく利用することができたか」という質問を行った．結果として，全体的にあまり好印象な結果を得ることができなかった．この結果から，システムの利用においてはストレスを感じるということが考えられる．その理由として，システム全体の処理時間の遅さがあげられる．システム全体の処理時間が遅いことで，ユーザーは待っている時間がいこと，またロード画面が静止画であるため，いつまで待てばいいのかがわかないことなどが考得られる．この解決策として，マルチプロセスや分散処理を用いたスクレイピングの更なる高速化や，分かち書きの高速化などがあげられる．また，3D グラフにおける描画処理も遅いため 3D グラフの描画手法についても検討が必要である．さらに，ロード画面に進捗バーなどを追加することで，処理が長くなってもあまりストレスなく利用することができると考える．

六個目に,「クラスターの提示は適切であるか」という質問を行った．結果として，肯定的な意見が三件，否定的な意見が一件，どちらでもないが一件となった．この結果から，入力するキーワードによって，出力されるクラスターが異なり，キーワードによってはあまり，適していないクラスターが出力されていることが考えられる．この理由として，今回用いたクラスタリング手法である k-means では外れ値による影響が多く，データによって

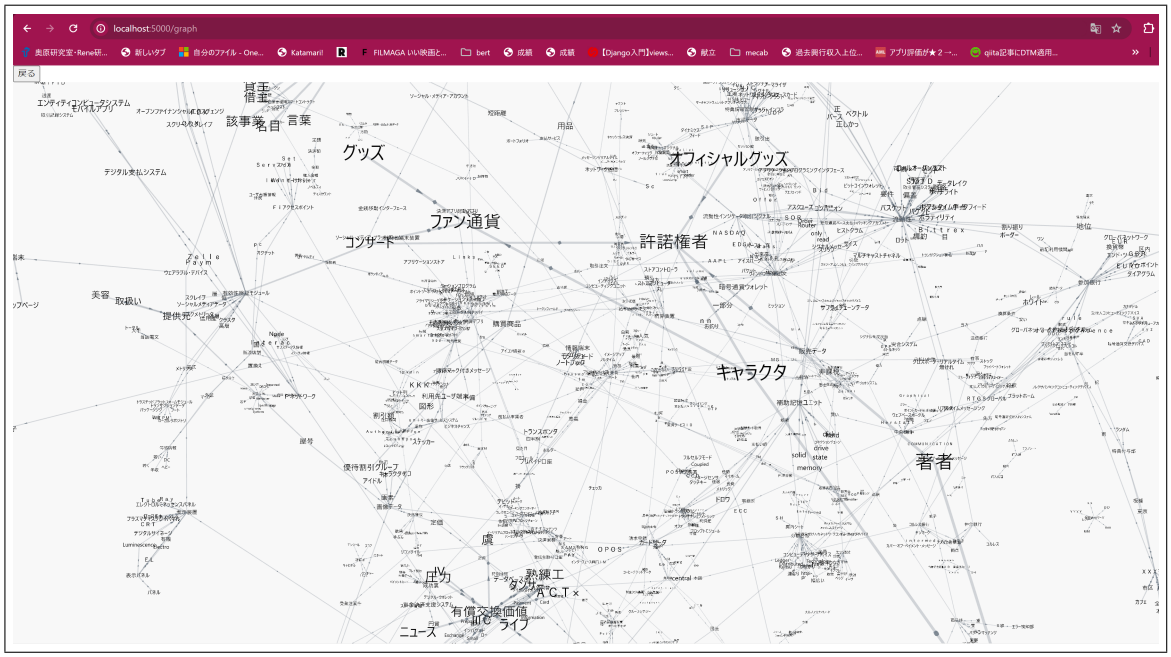


図 5.3: 出力された 3D グラフ

は，適していないクラスターが含まれる可能性がある．そこで，外れ値に強いクラスタリング手法を用いることで，これらの問題は解決すと考えられる．

七個目に,「共起語ネットワークは適切であるか」という質問を行った．結果として，肯定的な意見が三件，否定的な意見が一件，どちらでもないが一件となった．この結果から，入力するキーワードの違いや，取得されるデータの違いによって，共起語ネットワークの精度が異なることがあげられる．今回用いた simpson 係数でしきい値を設定したが，このしきい値が場合によってあまり適していないものであるということが考えられる．そこで，すべての場合において適するようなしきい値に変更することで解決できると考えられる．

八個目に,「3D グラフによる出力は適切であるか」という質問を行った．全体的に好印象な評価を得ることができた．この結果から，3D グラフによる共起語ネットワークの可視化は有用であるということがわかる．3D グラフで出力することで，よりインタラクティブなグラフになることが考えられる．

九個目に,「効率的な特許探索を行えそうか」という質問を行った．結果として，全体的に好印象な評価を得ることができた．この結果から，システムを用いずに行う特許探索よりも，システムを用いた特許探索の方が効率的であるということができる．特許全体を羅列するだけではなく，散布図による可視化や，共起語ネットワークによる可視化を行うことで，効率的な特許探索を行うことができると考える．

十個目に,「新しい知見を発見できそうか」という質問を行った．結果として，全体的に好印象な評価を得ることができた．この結果から，実際にシステムを利用することで，新しい知見を発見できると考えることができる．

また，自由記述では,「選択できる年数を増やした方がいい」という意見があり，入力されるキーワードによって，取得される特許の数が違い 24 年では十分な数の特許を取得することができなかったことが考えられる．そこで，もう少し取得する年数を増やすか，それらのキーワードが含まれる特許が多く含まれる年からのスクレイピングなどがあげられる．

表 5.2: アンケート結果

	解答者A	解答者B	解答者C	解答者D	解答者E
システムの操作性はわかりやすいか	4	4	5	4	4
システムの機能は理解しやすいか	3	5	4	5	4
レイアウトは適切か	4	4	5	4	5
デザインは見やすいか	5	4	4	5	5
ストレスなく利用することができたか	2	2	3	2	2
クラスターの提示は適切であるか	4	4	2	3	4
共起語ネットワークは適切であるか	3	4	5	2	5
3Dグラフによる出力は適切であるか	3	4	4	5	5
効率な特許探索を行えそうか	5	4	5	4	4
新しい知見を発見できそうか	4	5	5	4	4
入力してもらったキーワード	・スマホ ・キーホルダー	・アジ ・餌	・ネットワーク ・アローダイヤグラム	・音楽 ・楽曲 ・ゲーム	・アメリカ ・インド ・ドイツ

第6章

おわりに

本研究では，莫大な量の特許群を分析することで，IP ランドスケープ実施の支援を行うシステムの開発を行った．既存の特許プラットフォームでは，膨大な特許文献データを一気に集積し，特許全体をビッグデータとして分析を行うことは容易ではない．本システムでは，大量の特許文を効率的に収集し，特許情報を整理整頓し，そのうえでデータマイニングと機械学習の手法を駆使し，特許群から有用な知的財産情報を抽出，解析することを目的とした．このシステムを活用することで IP ランドスケープの調査や技術トレンド分析など，大規模な特許情報を活用した様々な業務支援を行った．

本研究で提案したシステムの特徴をまとめる．一つ目の特徴は，莫大な特許文章群をベクトル表現に変化し，そのベクトル空間上で潜在的なクラスタリングを行ったことである．現在までに蓄積された膨大な特許文章は，技術の進歩や新たな発明に伴い年々増加している．こうした文章群を一つの統一されたベクトル空間に投影することができれば，特許技術の全体像や内在する構造を可視化し，俯瞰的な解釈が可能になると考える．これらにより，従来になりマクロな視点から特許技術の全体を捉え，新たな知見の発見につなげることができることを確認した．

二つ目の特徴は，共起関係の分析による共起語ネットワークを作成しそれらを 3D グラフおよび 2D グラフによって可視化を行ったことである．2D グラフでは従来どおり共起語間の関係を平面上で表現することができる．2D グラフだけでなく 3D グラフによる描写によって，従来よりもより多くの情報を見ることができまた空間的な表現を行うことができる．これらのことにより，いままでの分析では得られなかった新たな知見をを得られることである．

今後の課題として，実行時間の短縮があげられる．本研究ではスクレイピングによる処理をマルチスレッドを用いることで高速化を図った．しかし，まだまだ処理の時間がかかっており更なる高速化が可能だと考えれらる．そこでマルチプロセスや GPU を用いた並列処理，他にも複数台のコンピュータを用いた分散処理などの手法が有効だと考えられる．さらに分かち書きの処理の高速化もあげられる．本手法で用いた分かち書きのモジュールである Janome はユーザー辞書の登録が容易であるのに対してデータの量が増えると処理時間が長くなるという問題もある．そこで近年開発された Vibrato のような高速な分かち書きシステムを用いることで高速に分かち書きを処理することができ使い勝手がよいシステムになると考える．以上の点を今後改善・検討することで，本手法の実用性と性能を一層向上させることができると考える．処理速度の向上こそが大規模データセットの分析では不可欠な要件であるといえる．

謝辞

本研究を遂行するにあたり，多大なご指導と終始懇切丁寧なご鞭撻を賜った富山県立大学工学部電子・情報工学科情報基盤工学講座の奥原浩之教授，António Oliveira Nzinga René 講師に深甚な謝意を表します．最後になりましたが，多大な協力をしていただいた研究室の同輩諸氏に感謝致します．

2024 年 2 月

平井 遥斗

参考文献

[1] 奥原 浩之, 尾崎 俊治, “適者生存型学習則を適用した競合動径基底関数ネットワーク”, 電子情報通信学会論文誌, pp. 3191-3199, 1997.

[2]

[3] 奥原 浩之, 佐々木 浩二, 尾崎 俊治, “環境の変化に適応できる複製・競合動径基底関数ネットワーク”, 電子情報通信学会論文誌, pp. 941-951, 1999.

[4]

[5] 倉本 和正, 鉄賀 博己, 東 寛和, 荒川 雅生, 中山 弘隆, 古川 浩平, “RBF ネットワークを用いた非線形がけ崩れ発生限界雨量線の設定に関する研究”, 土木学会論文集, pp. 117-132, 2001.

[6]

[7] 持田 英史, 飯國 洋二, “RBF ネットワークを用いた到来波方向の適応推定”, 電子情報通信学会論文誌, pp. 1205-1214, 2004.

[8]

[9] Silver, D., Huang, A., Maddison, C. J., Guez, A., Sifre, L., Van Den Driessche, G., Schrittwieser, J., Antonoglou, I., Panneershelvam, V., Lanctot, M., et al., “Mastering the game of go with deep neural networks and tree search ”, *Nature*, 529[7587]，2016.