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]．

本研究では，RC-RBFN が効率よく基底関数を削除あるいは追加するニューラルネットワークであることを示す．そして，機械学習における関数近似の手法として RC-RBFN を組み込んだのち，従来のアルゴリズムの場合と比較して有効性を確認することを目的とする．

§ 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 [4]．

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 [5]．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 [6]．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，正定関数 $V(\mathbf{w})$ として

$$V(\mathbf{w}) = \frac{1}{2} \int_{x \in B_{ik}} \{\eta(x) - s(x)\}^2 dx \tag{2.7}$$

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

$$s(x) = \sum_{j=1}^{M} w^j \xi^j(x) \tag{2.8}$$

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

$$\begin{aligned}\frac{dV(\mathbf{w})}{dt} &= \sum_{j=1}^{M}\frac{\partial V(\mathbf{w})}{\partial w^j}\frac{dw^j}{dt}\\ &= -\sum_{j=1}^{M}[\int_{x\in B_{ik}}\eta(x)\xi^j(x)dx - \sum_{h=1}^{M}\int_{x\in B_{ik}}\xi^j(x)\xi^h(x)dxw^h]\frac{dw^j}{dt}\\ &= -\sum_{j=1}^{M}w^j(\alpha^j - \sum_{h=1}^{M}\gamma^{jh}w^h)^2\\ &\leq 0 \end{aligned} \tag{2.9}$$

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

$$E(\mathbf{w}) = \frac{1}{2}\sum_{j}^{M}\{\eta(\mathbf{x}_j) - s(\mathbf{x}_j)\}^2 \tag{2.10}$$

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

$$\begin{aligned}\frac{dw^i}{dt} &= -\Delta\frac{\partial E(\mathbf{w})}{\partial w^i}\\ &= \Delta\sum_{x}\{\eta(x) - s(x)\}\xi^j w^j(x), \end{aligned} \tag{2.11}$$

$$\begin{aligned}\frac{dm_j}{dt} &= -\Delta\frac{\partial E(\mathbf{w})}{\partial m_j}\\ &= \Delta\sum_{x}\{\eta(x) - s(x)\}w^j\xi^j(x)\frac{(x-m_j)}{{\sigma_j}^2}, \end{aligned} \tag{2.12}$$

$$\frac{d\sigma_j}{dt} = -\Delta\frac{\partial E(\mathbf{w})}{\partial \sigma_i}$$
$$= \Delta\sum_x \{\eta(x) - s(x)\} w^j \xi^j(x) \frac{(x-m_j)^2}{{\sigma_j}^3} \tag{2.13}$$

が得られる．Δ は適当な正の定数である．

§ 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

$$\frac{dV(\mathbf{w})}{dt} = -\frac{V(\mathbf{w}^0)^R V(\mathbf{w})^{\frac{1}{r}}}{Rt^*} \tag{2.14}$$

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

$$\frac{dw^j}{dt} = \Delta(\alpha^j - \sum_{h=1}^{M} \gamma^{jh} w^h) w^j \tag{2.15}$$

とすることができる．このとき，Lyapunov 関数の時間変化は

$$\frac{dV(\mathbf{w})}{dt} = -\Delta \sum_{j=1}^{M} w^j (\alpha^j - \sum_{h=1}^{M} \gamma^{jh} w^h)^2 \tag{2.16}$$

となる．ここで，Δ は式〜と式〜から決定することができる．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

$$\frac{dw^j}{dt} = \frac{(\alpha^j - \sum_{h=1}^{M} \gamma^{jh} w^h) w^j}{\sum_{j=1}^{M} w^j (\alpha^j - \sum_{h=1}^{M} \gamma^{jh} w^h)^2} \times \frac{V(\mathbf{w}^0)^R V(\mathbf{w})^{\frac{1}{r}}}{Rt^*} \tag{2.17}$$

で定義することができる．

知的財産戦略とは企業が保有する知的財産を経営戦略の一環として取り入れ，企業の競争力を高め，事業目標を達成することを目的とする戦略である．知的財産には，特許，商標，意匠，著作権，ノウハウなど，さまざまな種類があり，これらの知的財産をどのように活用すれば，企業の価値を最大化できるのかを考えることが重要である [7].

知的財産戦略は，経営戦略と密接に関係しており，企業全体の戦略において各部門や機能の方向性を決定する重要な役割を果たしている．日本において，知的財産戦略は特許などの知的財産（Intellectual Property：IP）と景観や風景を意味する「Landscape」を組み合わせた造語で「IP ランドスケープ」と呼ばれることが多い．

知的財産戦略の目的は，以下の 3 つにまとめることができる [8].

（1）オープンイノベーション創出に貢献する知的財産戦略

- オープンイノベーションによる事業創出に貢献する知的財産戦略

  オープンイノベーションによる事業創出とは，近年の変化が激しい事業環境下において，従来のような社内のみで行う研究開発では，新規事業の創出に限界があることを踏まえ，競合企業やスタートアップ，大学等などの外部からの技術やアイディアを自社に取り組むこと等を通じて新たな価値を創造し，事業を創出しようとするもの．

- プラットフォーム戦略の推薦による事業創出に貢献する知的財産戦略

  プラットフォーム戦略の推進による事業創出とは，顧客や事業など，様々な主体を同一のプラットフォーム上に集めることで，事業のエコシステムを創出するビジネスモデルであるプラットフォーム戦略の推進により事業を創出しようとするものである．

- プソリューションビジネスの事業創出に貢献する知的財産戦略

  ソリューションビジネスとは，従来のモノ売りのビジネスから脱却し，顧客の課題を解決するコト売りへと進化したビジネスである．すなわちソリューションを創出するビジネスである．従来は知財部門が顧客の課題解決に直接関与することは少なかったが，近年は知財部門が積極的に関与し，新たなソリューションのコアを早期に特定し，これを適切に保護する知財ポートフォリオを構築している企業が増えている．

（2）事業競争力の強化に貢献する知的財産戦略

- コアインピーダンス強化に貢献する知的財産戦略

  コアコンピタンスとは，競合他社との差別化につながる競争優位性をもたらす自社の強みであり，これを技術として支えるのがコア技術である．コアコンピタンスを現状からさらに磨き，深化させることは，競争優位性を維持・強化するために重要である．

- グローバル事業展開に貢献する知駅財産戦略

  グローバル事業展開の形態として，輸出，ライセンシング，戦略的提携，買収及び現地子会社の新設等がある．

- M&A による事業ポートフォリオの拡大に貢献する知的財産戦略

  M&A による事業ポートフォリオ拡大とは，社外に存在する事業を M&A を実施して買収することで，自社の事業ポートフォリオを拡大することである．M&A は，既存の事業の規模拡大の経済効果や，新規事業への参入新たな技術やノウハウの獲得など，様々な目的で実施される．

（3）組織・基盤の強化に貢献する知的財産戦略

- ブランド価値向上に貢献する知的財産戦略

  ブランド価値の向上は，顧客からの信頼や好感を高め，他社に対しての競争優位性を構築するだけでなく，資金調達や人事確保の容易化など，企業の組織・基盤の強化にもつながる．ブランド価値は，高い経営理念に基づいた企業活動によって向上させることができる．

- デジタルトランスフォーメーション（DX) 等による事業基盤の強化に貢献する知的財産戦略

  デジタルトランスフォーメーションによる事業基盤の強化とは，IT やデータ等のデジタル技術をを活用して，自社の事業基盤の強化を図るものである．近年，知財情報等を自社の事業基盤を強化するために利用する取り組みが注目を集めており，DX において，知財部門が貢献できることは少なくない．

- SDGs への貢献に関わる知的財産戦略 SDGs（持続可能な開発目標）の取り組みは，国際社会から企業への信頼を高め，グローバルな投資家から高い評価を得るために重要である．また，企業の持続的発展のためにも欠かせないものとなりつつある．

IP ランドスケープでは，自社の経営・事業戦略を決める際に，経営・事業情報に知財情報を取り込んだ分析を実施する．その結果を経営者・事業責任者と共有し，結果に対するフィードバックを受けたり，立案検討のための議論や協議などを行う．

経営戦略

1960 年代，アメリカを中心に経営戦略論が台頭し始め，企業の成長と多角化への取り組みが盛んになった．組織と戦略の関係を提唱した Chandler は，戦略とは「企業が基本的な長期目的を決定して，これらの諸目的を遂行するために必要な行動方式を採択し，諸資源を割り当てること」と定義している．他にも，製品ー市場ミックスを提唱した Ansoff は，経営戦略とは「部分的無知のもとで，企業が新しい機会を探求するための意思決定ルール」と定義している [10].

経営戦略とは，会社が中長期的な目標を達成するために，資源の配分や事業内容の選択など，経営上の意思決定に関する長期的な方針である．経営戦略には，企業としての成長戦略や収益力強化戦略，事業ポートフォリオの見直しなどが含まれる．

効果的な経営戦略を立案するためには，自社の強みや特長を理解し，それを活かすことが重要である．また，事業環境の変化や競合他社の対応などを分析し，それに対応した戦略を練る必要がある．ステークホルダーの要望も踏まえた上で，最適な資源配分と事業の選択を行うことが大切である．

経営戦略の立案と実行は経営における最も重要な意思決定プロセスである．環境変化に対応しながら，戦略を実行することで，会社の目標達成と持続的成長を可能となる．このため，経営戦略には，トップの強い取り決めが不可欠であると言える．

経営戦略のフレームワーク

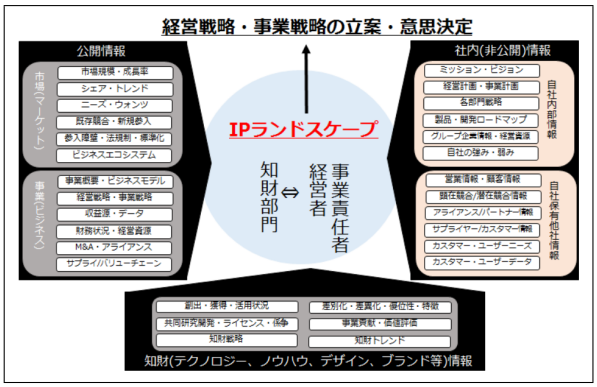


図 2.1: IP ランドスケープの概要 [9]

図 2.2: 特許文章の一例

経営戦略の策定では自社を取り巻く外部の環境要因に打ついて分析する外部環境分析や，自社内の環境を分析する内部環境分析を踏まえ自社の強みや弱み，機械や脅威を把握することで，戦略オプションを立案して最適な戦略を選択することが大切である．それらを行うために役立つ代表的なフレームワークとして，PEST 分析ファイブフォース分析，3C 分析，VRIO 分析，SWOT 分析，STP 分析，4P 分析などが挙げられる [11].

内閣府や特許庁による IP ランドスケープの積極的な推進に代表されるように，IP ランドスケープは研究機関においても積極的に検討されるべき対象であると考えられる．また，その具体的な取り組みの多くに ICT を活用した取り組みが多数行われていることから，IP ランドスケープの効率的な実施には ICT の活用が不可欠であり，情報工学との親和性が高いものと思われる．これらのことから，本研究は情報技術を用いた IP ランドスケープの支援を目的とする．

§ 2.3　基底関数の複製

一般に，未知の非線形関数を近似するために必要なニューロンの数があらかじめには不明である．そこで，冗長なニューロンを削除する手法が提案されている．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．

我々は先に，冗長なニューロンを削除できるシナプス可塑性方程式を導出し，これを適者生存型学習則としてシナプス結合荷重の更新則に適用した競合動径基底関数ネットワーク（Competitive Radial Basis Function Network：以下，CRBFN） を提案した．CRBFN

の特長は望ましい出力と相関が高い入力を伝達しているシナプス結合荷重が生き残り，生き残ったシナプス結合荷重に関係する入力ニューロンの入出力特性のみが調節されることである．そのため，CRBFN では競合により冗長なニューロンを消滅させることが可能であり，その結果，学習の高速化と過学習の回避が行われる．

しかしながら，CRBFN でも教師信号が変化するような環境の変化には対応しきれていなかった．その理由は CRBFN には新しい動径基底関数を追加する能力がないからである．ここでいう環境の変化とは入出力間の写像を与える関数そのものが変化する場合や，既に観測され学習に用いられていた入出力の組が不要となり取り除かれたり，新たに観測された入出力の組が学習に用いられたりするような変化などを想定している．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．この手法はシナプス可塑性方程式（Delta ルール，適者生存型学習則）に関する考察から得られるものであり，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，我々が先に提案した 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．

RBFN は非線形関数 $\eta(x)$ を動径基底関数 $\xi(x)$ の足し合せで近似するニューラルネットワークである．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．M 個の入力ニューロンと 1 個の出力ニューロンからなる RBFN は図 1 のような構造をもつ．d 次元の第 i 入力ベクトル $x_i \in R^d, (i = 1, 2, ..., N)$ はすべての入力ニューロンに入力される．第 j 入力ニューロン $(j = 1, 2, ..., M)$ はパラメータ ϕ_j をもつ．パラメータ ϕ_j は平均ベクトルと共分散行列の集合 $\{\mathbf{m}_j, \Sigma_j\}$ であるものとする．ここで，$\mathbf{m}_j = [m_i^1, m_i^2, ..., m_i^d]^{\mathrm{T}}$ であり，Σ_j はその逆行列 Σ_j^{-1} の第 kl 要素に σ_j^{kl} をもつ $d \times d$ の行列である．また，Σ_j は正定値対称行列である．第 j 入力ニューロンは入力ベクトル x_i に対して

$$\xi(\mathbf{x}_i, \phi_j) = exp\{-\frac{1}{2}(\mathbf{x}_i - \mathbf{m}_j)^{\mathrm{T}}\Sigma_j^{-1}(\mathbf{x}_i - \mathbf{m}_j)\} \tag{2.18}$$

を出力する．ここで，添字の T はベクトルの転置を示す．以後，このような出力を行う入力ニューロンのことを動径基底関数ということとする．出力値 $\xi(\mathbf{x}_i, \phi_j)$ はシナプス結合荷重 w_j を通して出力ニューロンへ伝達され，出力ニューロンでこれらは足し合わされ

$$s(\mathbf{x}_i, \mathbf{w}, \phi_j) = \sum_{j=1}^{M} w_j \xi(\mathbf{x}_i, \phi_j) \tag{2.19}$$

が出力される．ここで，$\mathbf{w} = [w_1, w_2, ..., w_M]^{\mathrm{T}} \in R^M$ であり，ϕ で集合 $\{\phi_1, \phi_2, ..., \phi_M\}$ を表す．ニューラルネットワークによる関数近似は，非線形関数 $\eta(x)$ をネットワークの出力 $s(\mathbf{x}, \mathbf{w}, \phi_j)$ で表すことである．そのため，RBFN による関数近似は累積 2 乗誤差関数

$$E(\mathbf{w}, \phi) = \frac{1}{2}\sum_{i=1}^{N} E(\mathbf{x}_i, \mathbf{w}, \phi) \tag{2.20}$$

の値を減少させることにより実現される．ここで，

$$E(\mathbf{x}_i, \mathbf{w}, \phi) = \{\eta(x_i) - s(\mathbf{x}_i, \mathbf{w}, \phi)\}^2 \tag{2.21}$$

は 2 乗誤差関数である．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 動径基底関数のシナプス結合荷重 w_j，パラメータ $\mathbf{m}_j$ 並びにパラメータ Σ_j である．

ここで，従来の RBFN と CRBFN の学習アルゴリズムの相違について述べる．一般の RBFN の学習アルゴリズムは式 (3) の累積 2 乗誤差関数に Delta ルール [7] を適用した

$$\Delta w_j = -\epsilon \frac{\partial E(\mathbf{w}, \phi)}{\partial w_j}, \Delta m_j^k = -\epsilon \frac{\partial E(\mathbf{w}, \phi)}{\partial m_j^k}, \Delta \sigma_j^{kl} = -\epsilon \frac{\partial E(\mathbf{w}, \phi)}{\partial \sigma_j^{kl}} \tag{2.22}$$

で与えられる．ただし，ϵ は適当な正の定数であり，m_j^k はパラメータ $\mathbf{m}_j$ の第 k 要素である．また，$\Delta w_j \equiv dw_j/dt, \Delta m_j^k \equiv dm_j^k/dt, \Delta \sigma_j^{kl} \equiv d\sigma_j^{kl}/dt$ である．ところで，未知の非線形関数を近似するために必要な動径基底関数の個数をあらかじめ知ることはできない．そのため一般に，RBFN では初期状態においていくつかの冗長な入力ニューロンを備えている．このことは，学習の遅延化や過学習を招く原因の一つであった．

CRBFN では，パラメータ mj 並びにパラメータ Σ_j の学習アルゴリズムは従来の RBFN と同じであり，Δm_j^k，$\Delta \sigma_j^{kl}$ により与えられる．しかし，シナプス結合荷重 w_{ik}^j に対しては Dale 則を考慮したシナプス可塑性方程式である適者生存型学習則

CRBFN の学習則は 2 乗誤差評価関数 $E(w)$ の値を減少させるために第 j 動径基底関数のシナプス結合荷重 w_j を式 (8) で更新する．学習中に $w_j \approx 0$ となった第 j シナプス結合荷重は消滅したものとして，生き残っているシナプス結合荷重，並びにそれらにより伝達される入力が変化を受ける動径基底関数のパラメータについてのみ学習を続ける．この学習則はシナプス結合荷重の更新則に特徴があるものの，平均ベクトルと共分散行列の更新則は従来の最急降下法を用いている．

そこで，本研究では更に CRBFN において平均ベクトルの更新則を改良することにより，動径基底関数を複製する競合動径基底関数ネットワークを新たに提案する．

3. パラメータが従う確率密度関数の導出ここでは，CRBFN の平均ベクトル，共分散行列とシナプス結合荷重が学習終了時にとる同時確率密度 $p(\mathbf{w}, \phi)$ を導出する．まず，シナプス結合荷重 w_j を

$$y_j{}^2 = w_j \tag{2.23}$$

と変数変換する．の定義域は任意の実数である．このとき，式 (8) は

$$\frac{dy_j}{dt} = (\frac{\alpha_j}{2} - \sum_{k=1}^{M} \frac{\gamma_{jk}}{2} y_k^2) y_j \tag{2.24}$$

となる．ただし，ϵ は省略した．式 (14) は積分条件

$$\frac{\partial}{\partial y_k} \frac{dy_j}{dt} = \frac{\partial}{\partial y_j} \frac{dy_k}{dt} \tag{2.25}$$

を満たすためポテンシャル

$$V'(\mathbf{y}) = -\sum_{j=1}^{M} \int_a^{y_j} (\frac{\alpha}{2} - \sum_{k=1}^{M} \frac{\gamma}{2}) y_j' dy_j' \tag{2.26}$$

を考えることができ，変数 y_j の時間変化はポテンシャル $V'(\mathbf{y})$ から，

$$\frac{dy_j}{dt} = -\frac{\partial V'(\mathbf{y})}{\partial y_j} \tag{2.27}$$

で導くことができる．関係式 (13) からポテンシャル $V'(\mathbf{y})$ は

$$V(\mathbf{w}) = -\sum_{j=1}^{M}\{\frac{\alpha}{4} - \sum_{k\neq j}^{M}\frac{\gamma}{4}w_j w_k - \frac{\gamma}{8}w_j^2\} \tag{2.28}$$

と書き直すことができる．この結果,

$$E(\mathbf{w}) = 4V(\mathbf{w}) + \frac{1}{2}\sum_{i=1}^{N}\eta^2(x_i) \tag{2.29}$$

であることが示されるので，累積 2 乗誤差関数 $E(\mathbf{w})$ の最小化はポテンシャル $V(\mathbf{w})$ の最小化と等価であることがわかる．

今，式 (14) に従う y_j はポテンシャル $V'(\mathbf{y})$ 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

$$y_j(t+\Delta t) = y_j(t) - \frac{\partial V(\mathbf{y})}{\partial y_j}\delta t + \sqrt{Q\Delta t}n_j(t) \tag{2.30}$$

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 の平均ベクトル，共分散行列とシナプス結合荷重が満たす同時確率密度 $p_\beta(\mathbf{w})$ は

$$p_\beta(\mathbf{w}) = Z_\beta^{-1}exp\{-\beta V(\mathbf{w})\} \tag{2.31}$$

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

$$Z_\beta = \int_{\mathbf{w}}\int_{\phi} exp\{-\beta V(\mathbf{w},\phi)\}d\mathbf{w}d\phi \tag{2.32}$$

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 (19) より

$$p_{\beta'}(\mathbf{w}) = Z_{\beta'}^{-1}exp\{-\beta' E(\mathbf{w})\} \tag{2.33}$$

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．シナプス可塑性方程式として Delta ルールを用いている従来の RBFN に関しても，同様にしてパラメータが従う確率密度関数が導出できることを付録で示す.

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

$$\eta(x) = 3N(-1.5,1) + 2N(1,0.5) \tag{2.34}$$

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

$$p_{\beta'}(m|w, \Sigma) = \frac{p_{\beta'}(w)}{\int_m p_{\beta'}(w)dm} \tag{2.35}$$

は導出することができ，それは図 2 のようになる．

この結果から，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 $E(\mathbf{w})$ 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 $E(\mathbf{w})$ を小さくできることもわかる．式 (25) を更に，シナプス結合荷重 w とパラメータ Σ で積分をとれば確率 $p_{\beta'}(m)$ が算出できる．そこで，教師信号を復元するために確率 $p_{\beta'}(m)$ を極大とするパラメータ m に動径基底関数を配置するような一撃アルゴリズムを考えることもできる．しかしながら，多次元の場合にはシナプス結合荷重 w とパラメータ Σ の積分が困難であることから，本研究では確率 $p_{\beta'}(m)$ を用いずに，条件付き確率を用いて逐次的にパラメータ m を求めていく方法を考える．

4. 動径基底関数の複製アルゴリズム 4. 1 自由エネルギーの導出一般に，式 (6) に従いパラメータ m_j^k を更新し続けると極小解にとらわれ，累積 2 乗誤差関数 $E(\mathbf{w})$ 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 $E(\mathbf{w})$ 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 $E(\mathbf{w})$ 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 x_i をパラメータ m_i とする動径基底関数を作成することができる．そこで，この手法を CRBFN に組み入れたニューラルネットワークを複製・競合動径基底関数ネットワークと呼ぶこととする．

ところで，累積 2 乗誤差関数 $E(\mathbf{w})$ の最小化は各入力ベクトル x_i ごとに 2 乗誤差関数 $E(\mathbf{x}_i, \mathbf{w})$ を最小化することに等価である．そこで，各入力ベクトル 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 x_i の条件付き確率密度関数は

$$p_{\beta'}(\mathbf{x}_i|\mathbf{m}_{j[i]}, \phi'_j, \phi''_j) = Z_{\beta'}^{-1}(\mathbf{m}_j, \phi'_j, \phi''_j) exp\{-\beta' E(\mathbf{x}_i, \mathbf{m}_{j[i]}, \phi'_j, \phi''_j)\} \tag{2.36}$$

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

$$Z_{\beta'}(\mathbf{m}_j) = \sum_{i=1}^{N} exp\{-\beta' E(\mathbf{x}_i, \mathbf{m}_{j[i]})\} \tag{2.37}$$

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

$$\langle E(\mathbf{m}_j)\rangle_{\beta'} = \sum_{i=1}^{N} p_{\beta'}(\mathbf{x}_i|\mathbf{m}_{j[i]}) E(\mathbf{x}_i, \mathbf{m}_{j[i]}) \tag{2.38}$$

が一定となるという二つの制約のもとで，エントロピー

$$\langle E(\mathbf{m}_j)\rangle_{\beta'} = \sum_{i=1}^{N} p_{\beta'}(\mathbf{x}_i|\mathbf{m}_{j[i]}) E(\mathbf{x}_i, \mathbf{m}_{j[i]}) \tag{2.39}$$

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

$$F_{\beta'}(\mathbf{m}_j) = -\frac{1}{\beta'} log Z_{\beta'}(\mathbf{m}_j) \tag{2.40}$$

で定義すれば,

$$S_{\beta'}(\mathbf{m}_j) = -F_{\beta'}(\mathbf{m}_j) + \beta' \langle E(\mathbf{m}_j)\rangle_{\beta'} \tag{2.41}$$

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^k の更新則を式 (6) の Δm_j^k の代わりに

$$\Delta_{\beta'} = -\epsilon \sum_{i=1}^{N} \frac{\partial F_{\beta'}(\mathbf{m}_j)}{\partial m^k_{j[i]}} \tag{2.42}$$

$$= -\epsilon \sum_{i=1}^{N} \frac{\partial F_{\beta'}(\mathbf{m}_j)}{\partial Z_{\beta'}(\mathbf{m}_j)} \frac{\partial Z_{\beta'}(\mathbf{m}_j)}{\partial m^k_{j[i]}} \tag{2.43}$$

$$= \sum_{i=1}^{N} p_{\beta'}(\mathbf{x}_i|\mathbf{m}_{j[i]}) \Delta m^k_{j[i]} \tag{2.44}$$

$$= \langle \Delta m^k_j \rangle_{\beta'} \tag{2.45}$$

で与えることとする．ここで，

$$\Delta m^k_{j[i]} = -\epsilon \frac{\partial E(\mathbf{x}_i, \mathbf{m}_{j[i]})}{\partial m^k_{j[i]}} \tag{2.46}$$

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

$$\Delta_0 m^k_j = \Delta m^k_j \tag{2.47}$$

であることが示される．この場合は，RC-RBFN のパラメータ m^k_j の更新則が従来の RBFN のパラメータ m^k_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，$\Delta_0 m^k_j = 0$ とするパラメータ $m_{j[i]}$ は

$$\sum_{i=1}^{N} \xi(\mathbf{x}_i, \mathbf{m}_{j[i]})(\mathbf{x}_i - \mathbf{m}_{j[i]})\{\eta(x_i) - s(\mathbf{x}_i, \mathbf{m}_{j[i]})\} = 0 \tag{2.48}$$

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．得られた結果を確認するために，式 (24) の教師信号から適当に 5 点選んだのが表 1 である．これを教師入力信号 $x_i(i = 1, 2, ..., 5)$ とする．式 (32) において β' を 0，パラメータ Σ を徐々に小さくしたときに，$\Delta_0 m^k_j = 0$ となる点をプロットしたのが図 3 である．パラメータ $\Sigma \to 0$ において $m_{[i]} = x_i(i = 1, 2, ..., 5)$ へ収束していることがわかる．ただし，この例では $j = k = 1$ であるため，添字の j と k は省略した．あるいは，逆にパラメータ Σ_j を固定したまま $\beta' \to \infty$ にすると，$\Delta_\infty m^k_j = 0$ とするパラメータ $m_{j[i]}$ は

$$\sum_{i=1}^{N} exp\{-\beta' E(\mathbf{x}_i, \mathbf{m}_{j[i]})\}\xi(\mathbf{x}_i, \mathbf{m}_{j[i]})(\mathbf{x}_i - \mathbf{m}_{j[i]})\{\eta(\mathbf{x}_i) - s(\mathbf{x}_i, \mathbf{m}_{j[i]})\} = 0 \tag{2.49}$$

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^k_j の更新則 $\Delta_{\beta'} m^k_j$ では，$\Delta_0 m^k_j$ で従来の RBFN のパラメータ m^k_j の更新則を実現し，更に，$\Sigma_j \to 0$ とすればすべての入力ベクトル $\mathbf{x}_i$ の第 k 要素 $x^k_i(\forall i)$ をパラメータ m^k_j の安定な収束点として検出できることがわかる．あるいは，$\Delta_\infty m^k_j$ 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

$$F_{\beta'} = -\frac{1}{\beta'} log Z_{\beta'} \tag{2.50}$$

を考える．ただし，分配関数は

$$Z_{\beta'} = \sum_{i=1}^{N} \sum_{j=1}^{M} exp\{-\beta' E(\mathbf{x}_i, \mathbf{m}_{j[i]})\} \tag{2.51}$$

で与えられる定数である．このとき，式 (30) は

$$\begin{aligned} F_{\beta'}(\mathbf{m}_j) - F_{\beta'} &= -\frac{1}{\beta'} log \frac{Z_{\beta'}(\mathbf{m}_j)}{Z_{\beta'}} \\ &= -\frac{1}{\beta'} log \sum_{i=1}^{N} p_{\beta'}(\mathbf{x}_i, m_{j[i]}) \end{aligned} \tag{2.52}$$

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

$$p_{\beta'}(\mathbf{x}_i|\mathbf{m}_{j[i]}) = Z_{\beta'}^{-1} exp\{-\beta' E(\mathbf{x}_i, \mathbf{m}_{j[i]})\} \tag{2.53}$$

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^k を式 (32) の $\Delta_0 m_j^k$ 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^k を更新する．

STEP 4. 分岐により $\Delta_{\beta'} m_j^k = 0$ 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 $m_p^n(n \neq k)$ 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^k は新たに増えた点とする．STEP 1. へ戻る．

特許情報とは，特許・実用新案・意匠・商標の出願や権利化に伴って生み出される情報である．この情報は，研究開発の重複防止，既存技術の活用，無用な紛争の回避などに役立つ．特許情報は，研究開発の策定から商品化，更には他人の権利調査に至るまでの様々な事業活動において活用されている．

特許の一例

本研究で使用する特許の一例を図 2.2 に示す．このように特許はタイトルと要約である Abstract，国際特許分類（International Patent Classification）という特許分類を示す Classification，本文を示す Desciption，請求項を示す Claims，そして特許自体の ID と出願日などの情報を含んだ部分からなる．

具体的な活用例は，以下のとおりである．[12].

特許情報の活用例

- 技術動向調査

  将来性を見据えた研究テーマの選定や過去になされた研究との重複回避のために，特許情報を利用して技術動向調査が行われる．特定の技術分野における特許出願の動向や出願件数の推移を調査することにより，過去にどのような技術が存在したか，また，今後開発すべき技術分野の把握の参考になる．

- 出願前の先行技術調査

  研究成果として発明がなされたとき，権利化するか否かの判断が必要となる．特許出願をする際に関連する分野の先行技術について調査することにより，権利として認められる見込みのない無駄な出願を未然に防止することができる．

- 権利調査

  開発製品が他人の産業財産権を侵害すると，製造・販売の中止や製造品の廃棄，あるいは権利者への損害賠償にまで発展する恐れがある．これらを未然に防止するために，設計から製造前段階にかけて，他人の権利範囲の調査を行う．

- 公知例調査

  他の権利者から警告を受けた場合などの対抗手段として，自社の発明・考案を事業化する際に障害となる他人の特許権・実用新案権を無効にするため，その特許・実用新案登録の出願前の公知例を調査する．

- 公知例調査

  事業を営む上で多くの場合には競合他社が存在している．その競合他社がどのような戦略で事業を行っているか調査する上で，特許情報は貴重な情報源となる．競合他社の過去から現在に至るまでの出願動向を把握することにより，研究開発動向等を読み取ることが可能である．また，競合他社の出願動向を継続的に監視し，自社にとって障害となる出願等の早期発見に努めることも重要である．

特許番号

特許番号とは，特許として認められた発明に付与される 7 桁の番号である．特許番号は「特許代 XXXXXXX 号」のように表記されている．特許番号は，原則として，出願，公開，登録の審査段階それぞれで付与され，審査段階，年，通し番号で構成されている．この特

図 2.3: 検索結果の例

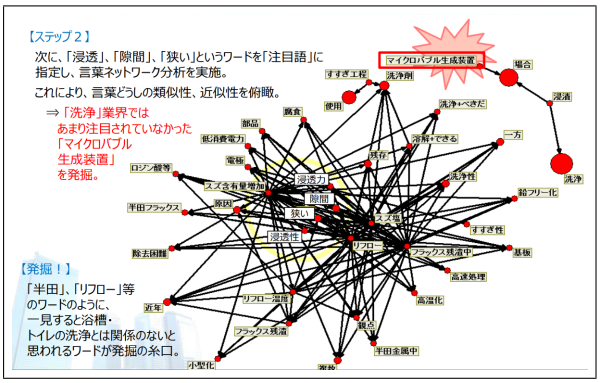


図 2.4: 言葉ネットワーク [13]

許番号は発明の特定性と公示性の観点から非常に重要なものであり，特許権の貴族や特許文献検索の際の必須情報となっている．また特許関連書類の請求や特許料金の納付の際にもこの番号を提示する必要がある．特許番号は，その発明を社会に対して公表し保護しているという証であり，特許における根幹として機能している．特許番号は，特許情報の管理と分析に重要な基盤となっている．具体的には，特許番号をキーに特許プラットフォーム上で技術動向を探索，照会したりすることができる．

特許プラットフォーム

特許プラットフォームとは，特許情報データベースを提供するオンラインサービスのことである．主の特許プラットフォームとしては，各国の特許庁が運営するデータベース，Google Patents，商用データメーカーが提供するサービスなどがある．主要なプラットフォームでは，出願番号から基本情報の検索や概要文の参照が可能である．また，発明者，出願日範囲，国別分類コードなどから検索フィルターを掛けることもできる．特許プラットフォームにおける検索結果は通常，特許を一覧形式で表示するケースが多い．特許プラットフォームである Google Patents におけるの検索結果を図 2.3 に示す．実際に自分が知りたいものを探すためにはそれらの中から自力で見つけ出す必要がある．現在の特許プラットフォームでは特許全体をビッグデータとして扱いたい場合には適しているとはいいがたい．

特許情報処理

特許情報処理とは，特許文章や特許データベースの情報をコンピュータで自動的に解析・処理を行う技術のことである．その目的は，莫大な量の特許情報から有用な知見を効率的に見出し，知的財産戦略の立案や技術動向の分析などに活用する．具体的には，自然言語処理やテキストマイニング，情報検索，データマイニングといった手法を用いて，特許文章の意味内容を理解したうえで重要なキーワードを抽出したり，文章間の関連性を判断したりする．また，抽出したデータを可視化し特許マップを作成しることも一般的である．特

許情報処理の利用目的としては，特許可否判断の支援，先行技術調査の効率化，技術開発のトレンド分析，競合他社の特許戦略の把握などがあげられ，特許業務の生産性向上や質の向上に役立つテクノロジーといえる．以下に実際の活用事例を紹介する [13].

IPL への活用事例

＜浴室・トイレの室内洗浄技術の課題を解決する技術を探る＞

ステップ 1

洗浄に関する特許出願の【発明が解決しようとする課題】に記載された文章をテキストマイニングし，係り受け関係のある言葉の頻度をランキング．これによれば，頻度が高く修飾/被修飾関係にある言葉のペアは,「効率-良い」洗浄,「狭い-隙間」部分の洗浄等が,「洗浄」に関する多くの出願が解決使用とする課題であることがわかる．

このことから,「洗浄」に関する業界では，安全性が高く，狭い隙間にも浸透する洗浄力の高い洗浄剤・洗浄方法が模索されていることがわかる．

ステップ 2

次に,「浸透」,「隙間」,「狭い」というワードを指定し，言葉ネットワーク分析を実施．これにより，言葉同士の類似性，近似性を俯瞰．実際の言葉ネットワークを図 2.4 に示す．このことから「洗浄」業界ではあまり注目されていなかった「マイクロバブル生成装置」を発掘

ステップ 3

特許出願の【発明が解決しようとする課題】に記載されたワード「浸透力」に着目し,「浸透」,「狭い」,「隙間」等のワードを含む発明を抽出したところ，25 件がヒット．

ステップ 4

数十ナノメートルという極めて小さな気泡は，ウルトラファインバブル (UFB) と呼ばれる．UFB は，透明で視認できないことに加え，その気泡が極めて長期間 (数ヶ月) 液中に存在しうることや，気泡が電荷を帯びること，気泡内部が超高圧状態になること等の特異な特性がある．産業界では，その特性を利用した UFB の応用が幅広い分野で検討されている．たとえば，食品分野をはじめとして，化粧品，薬品，医療，半導体や植物育成等，幅広い分野での応用がさかんに考えられており，ウルトラファインバブルに大きな期待．

上記のように，特許情報は，多岐にわたる分野で活用されており，その重要性は年々増している．これらの特許情報から生み出されるものの価値を高めるためには，データ解析技術を活用することが重要である．特許情報処理を用いた分析から得られる知見を IP ランドスケープに取り入れることが期待されており，今後もますます，重要性を増すと考えられる．

第3章

特許情報の可視化

§ 3.1　特許情報のベクトル化

特許情報は，日々蓄積され，今では莫大な量となっており，それらの分析は困難を極める．そこで，特許情報を効率的に分析するためには，各特許をベクトル化して整理をおこない，全体を俯瞰できるように可視化する必要があると考える．本研究では，特許本文の文章を対象にベクトル化を行う．特許本文には，特許技術の内容が詳細に記載されているため，これらの情報をベクトル化することで，特許の技術分野や技術トレンドなどを把握することができると考える．

具体的には，特許本文を Sentence-Bidirectional Encoder Representation from Transform（Sentence-BERT）を用いることで文章全体を単位にベクトル化を行う [18]．Sentence-BERT は，Bidirectional Encoder Representations from Transformers（BERT）をベースに開発されており，文章の単語の順序を考慮して，文章の意味を表現するベクトルを生成する．

sentence-BERT は，文章の意味を理解する能力に優れているため，自然言語処理の様々なタスクに活用されている．

BERT

BERT は，Google が提案した最新の言語モデルの一つであり，BERT のキーポイントは，Encoder のみの Transformer アーキテクチャを採用し，Attention メカニズムを用いて単語間の関係性をモデル化している．BERT による処理の流れを図 3.1 に示す．BERT の目的は，あるテキストから単語や語句の意味表現を文章全体の文脈に依存したリッチなベクトル表現で表現することである．これは下流のテキスト分類や質問応答などに有用な汎用的な言語表現を獲得できることを意味する．BERT は Masked LM と次文予測の 2 つのタスクで事前学習を行うことで統計的な言語モデルを構築している．Masked LM ではランダムにマスクした単語を予測することで文章理解能力を高め，次文予測では文章間の関係性を学習している．この事前学習済みモデル BERT モデルは，下流タスクの比較的小さいデータセットで微調整することで転移学習が可能である．結果として，多くの NLP タスクで従来手法を上回る精度を達成しており，BERT は分散表現と転移学習において大きな進展をもたらした．

Transformer

近年，翻訳などの入力文章を別の文章で出力するというモデルは，Attention を用いたエンコーダー，デコーダ形式の RNN や CNN が主流であった．しかし，Transformer は，RNN

や CNN を用いず Attention のみを用いたモデルである．Transformer は，再帰も畳み込みも一切行わないので並列化が容易であり，他のタスクにも汎用性が高いという特徴がある．Transformer においては Attention を多数並列に配置した Multi-Head Attention が用いられ，一般的に以下の式（3.1）のように定式化される [17].

Multi-Head Attention

$$Multi\text{-}HeadAttention(Q, K, V) = Concat(head_1, head_2, \cdots, head_h)W_o \tag{3.1}$$

$$\text{where } head_i = ScaledDotProductAttention(QW_i^Q, KW_i^K, VW_i^V) \tag{3.2}$$

ここで，Scaled Dot-Product Attentioin では，内積を利用したベクトル間の類似性に基づく変換を行われ，一般に以下の式（3.3）のように定式化される．

<Scaled Dot-Product Attention>

$$Attention(Q, K, V) = softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)V \tag{3.3}$$

3.1 では，学習パラメータを持っていない Scaled Dot-Product Attention の表現力を広げるために，入力の直前に学習パラメータを持つ Linear 層の追加を行っている．これにより，入力されるベクトルの特徴空間に依存しない注意表現を学習することができる．Linear 層の追加を行った Scaled Dot-Product Attention を一般に Single-Head Attention を呼ぶ.

<Attention への入力方法 >

Scale Dot-Product Attention は，ある単語に対して，その単語が文章に含まれる単語とどれだけ類似しているのかを計算し，それらを確率的に表現したものである．Transformer における Attention の入力には主に以下の 2 種類の入力方法が用いられている．

1. Self-Attention（softmax に与える Query，Key，Value を同じ値にする）
2. SourceTarget-Attention（Key，Value を同じ値にし，Query を異なる値にする）

Single-Head Attention では多種多様な意味や文法をもつ単語に対しても単一の注意表現が生成される．そこで，Single-Head Attention を多数並列に配置して Multi-Head にすることで，複数の特徴部分空間における注意表現の獲得をすることができる．

以上のことから，文章を行列で表せることが分かった．しかし，文章というのは，文字を読む方向が重要であり，行列として表され，かつ，一括で処理する場合，文字の順番の概念がなくなってしまう．このことが原因となり，文章を正しく扱えなくなる可能性がある．そのため，Embedding 層からの行列に位置情報を含んだ行列を足し合わせることで，文字の順番の概念を扱えるようにする必要がある．これを可能にするのが Positional Encoding である．

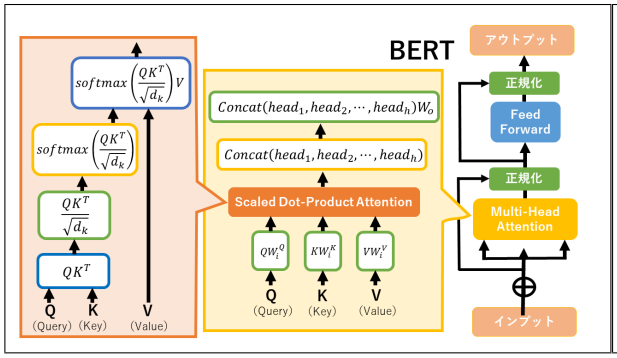


図 3.1: BERT による処理の流れ

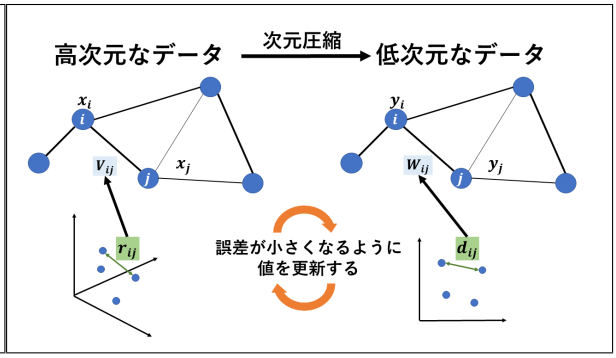


図 3.2: UMAP による次元圧縮

Positional Encoding

$$PE_{(pos,\,2i)} = sin\left(\frac{pos}{10000^{\frac{2_i}{d_{model}}}}\right) \tag{3.4}$$

$$PE_{(pos,\,2i+1)} = cos\left(\frac{pos}{10000^{\frac{2_i}{d_{model}}}}\right) \tag{3.5}$$

入力文章の単語数が 50 個まで扱えて，Embedding 層の埋め込み次元数が 128 次元の場合，Positional Encoding が生成する行列は 128 次元の行ベクトルが縦に 50 個並んだ行列になる．この行列は，各行のベクトルが絶対に同じものにならいため，この行列から単語の位置情報を表すことができる．具体的には，行ベクトルの各次元は，単語の位置情報に応じて，異なる値が割り当てられている．例えば，最初の行ベクトルの最初の次元は，単語の位置が 0 であることを示し，最後の行のベクトルは，単語の位置が 49 であることを示す．このように，Positional Encoding は，単語の位置情報を行ベクトルに埋め込むことで，Transformer モデルが単語の順序情報を利用でいるようにしている．

Sentence-BERT

BERT では 2 つの文章を入力し，それらの類似度を測ることができる．しかし，複数の文章を入力する場合は BERT では容易ではない．そこで本研究では Sentence-BERT を用いる．

Sentence-BERT ではファインチューニングを行っている．具体的には「Siamese Network」というものを使い，2 つのニューラルネットワークを用いて，それぞれの埋め込み表現を計算し，その埋め込み表現についての比較を行っている．BERT で求められた埋め込み表現を時系列方向に pooling し，それらを Softmax 関数を用いて，分類を行っている [19].

<Siamese Network>

$$O = softmax(W_t(u, v, |u - v|)) \quad W_t \in R^{3_n \times k} \tag{3.6}$$

事前学習モデルは Hugging Face や GitHub などのサイト公開されている．また東京大学や京都大学なども独自のモデルを公開している．本研究では Hugging Face に登録されている ” sonoisa/sentence-bert-base-ja-mean-tokens ” を用いる．

§ 3.2　次元圧縮手法とクラスタリング手法

今回扱うデータは 768 次元と高次元であるためクラスタリングを行う際に次元の呪いが発生することが考えられるため，次元圧縮を行う．次元圧縮手法には線形次元圧縮手法と，非線形圧縮手法がある．線形次元圧縮手法は，計算がようであるが，データの非線形的な構造を表現することが難しい．一方で，非線形次元圧縮手法は，データの非線形的な構造を表現することができるが，計算が複雑で，処理に時間がかかる．今回行う次元圧縮では，ベクトル同士の近さを保持する必要である．ベクトル同士の近さを保持するためには，非線形次元圧縮を用いる必要がある．そこで本研究での次元圧縮手法には UniformManifold Approximation and Projection of DimensionReduction（UMAP）を用いる．

UMAP

UMAP は 2018 年に，Leland Mclnnes，John Hecht，James Melville によって提案された手法である [20]．従来の非線形次元圧縮手法である t-SNE よりも実行時間が高速であり，圧縮後の情報保持力が高いという特徴がある．また，4 次元以上にも圧縮することが可能である．UMAP は高次元空間での近いデータ同士を低次元空間でも近く，異なる点同士を遠ざけるという処理を行うこと，で高次元でのデータ同士の近さや遠さを低次元でも表現できるようにしている．図 3.2 に UMAP による次元圧縮のイメージを示す．

点 x_i に対して，k 番目に距離の小さい点までを集めた集合である k 近傍の集合を K_i とあらわす．この時，他のある点 x_j が集合 K_i に属すか否かは，$0, 1$ の 2 値で表現可能である．UMAP では，この 2 値を，0 以上 1 以下の実数に拡張した，ファジー集合として扱う．高次元空間におけるデータ同士の近さは以下のように定義され，3.9 によって対称化される [21]．

重み付き k 近傍（高次元）

$$v_{j|i} = exp\left(\frac{-(r_{ij} - \rho_i)}{\sigma}\right) \tag{3.7}$$

$$\rho_i = min_{j \in K}\{r_{ij}\} \tag{3.8}$$

式 3.7 において，σ_i は，点が密集しているところでは小さく，疎なところでは広く設定する変数であり，この変数は $\sum_j v_{j|i} = log_2 k$ となるように 2 部探索でも求められる．

＜対称化＞

$$v_{ij} = (v_{j|i} + v_{i|j}) - v_{j|i}v_{i|j} \tag{3.9}$$

低次元空間におけるデータ同士の近さは以下のように定義され，3.11 によって対称化される．

重み付き k 近傍（低次元）

$$w_{ij} = exp(-max\{0, d_{ij} - \rho'\}) = \tilde{w}_{ij} \tag{3.10}$$

＜対称化＞

$$w_{ij} = \frac{1}{1 + a \cdot d_{ij}^{2 \cdot b}} \tag{3.11}$$

トポロジカル表現の最適化によって高次元空間での近さと低次元空間での近さができるだけ同じになるように最適化を行う.

トポロジカル表現の最適化

$$L = \sum \left[v_{ij} log \frac{v_{ij}}{w_{ij}} + (1 - v_{ij}) log \frac{1 - v_{ij}}{1 - w_{ij}} \right] \tag{3.12}$$

UMAP によって次元圧縮を行ったデータに対してクラスタリングを行い，潜在的なグループ化を行う.

クラスタリング手法

＜階層型クラスタリング＞

データセット内の観測地を木構造（階層構造）組織化するクラスタリング手法である．この手法では，最初はデータ点を単一のクラスタとみなし，類似性が高いデータ点を結合していき，最終的に全体が一つのクラスタになるまで続けられる.

＜分割型クラスタリング＞

分割型クラスタリングは，データセットを複数のクラスタに分割するクラスタリング手法である．この手法では，データが異なるクラスタに属するように分割され，各クラスターはほかのクラスとは異なる特徴や属性を持つ.

＜確率モデル型クラスタリング＞

データセットのクラスタ構造を確率モデルを用いてモデリングするクラスタリング手法である．データが生成されるプロセスを確率分布としてとらえ，それに基づいてクラスタリングを行う.

本研究では，クラスターの重心を求める必要があるためクラスタリングには k-medoids を用いる [?]．k-medoids は k-means よりも外れつに強いクラスタリング手法である.

K-means

n 個の個体を $\vec{x_i} = (x_{i1}, \ldots, x_{iD}), i = 1, \ldots, n$ で表し，この個体の集合を X とする．データを K 個の重なりの無いクラス $X_k, k = 1, \ldots, K$ に分類するため，次の目的関数を用いる.

$$J = \min_{\{\vec{c_k},\, k=1,\cdots,\, K\}} \sum_{i=1}^{n} \sum_{\vec{x} \in X_k} \|\vec{x_i} - \vec{c_k}\|^2 \tag{3.13}$$

ここで，$\|\vec{x_i} - \vec{c_k}\|^2 = \sum_{d=1}^{D} (x_{id} - c_{kd})^2$ とし，クラスター中心は $\vec{c_k} = (c_{k1}, \ldots, c_{kD}$ である．式について最小化を行うため，K-Menas アルゴリズムと呼ばれる以下の反復アルゴリズムを使用する.

1. $\vec{c_k}^t$ が与えられたとき，それぞれの $\vec{x_i}$ に関して次式を計算する．

$$a = \underset{k}{argmin} \|\vec{x_i} - \vec{c_k}^{(t)}\|^2 \tag{3.14}$$

2. $X_k^{(t)}$ が与えられたとき，次式を計算する．

$$\vec{c_k}^{t+1} = \frac{1}{n_k^{(t)}} \sum_{\vec{x_i} \in X_k^{(t)}} \vec{x_i}, k = 1, \ldots, K \tag{3.15}$$

ここで，$n_k^{(t)}$ は $X_k^{(t)}$ に属する個体の数であり，$\vec{c_k}^{t+1}$ は $X_k^{(t)}$ の $(k+1)$ 回目のクラスター中心でありクラスターの代表点に相当する．
ある小さい定数を ϵ とし，すべての k について終了条件である $\|\vec{c_k}^{t+1}\vec{c_k}^t\| < \epsilon$ を満たすまで，(1) と (2) の手順を繰り返す．

K-medoids

k-medoids は，k-means アルゴリズムの改良版であり，k-meansa では，データセットを k 個のクラスタに分割し，各クラスタに代表点を割り当てる．k-means では外れ値が存在すると，代表点の位置が大きく変化し，クラスタリングの結果に影響を受ける．k-medoids では外れ値が存在しても，その影響を軽減することができる．しかし，k-medoids は k-means に比べて計算量が増加する．

k-medoids と k-means の違い

- k-means：各クラスタの代表点は，そのクラスタ内のデータの平均値となり，クラスタ内のデータの平均を計算し，その平均を代表点とする．
- k-medoids：各クラスタの代表点は，そのクラスタの別の点となり，クラスタ内のデータの中から，他のデータ点との総距離が最小となるデータ点を代表点とする．

K-medoids を行うには初めにクラスター数を与える必要があるため，シルエット分析を用いて最適なクラスター数を決定する．

シルエット分析

シルエット分析はクラスタ内は密に凝集されているほど良い．異なるクラスタは花らているほど良い．この二つをもとに最適なクラスタ数を求める．各データサンプル $x^{(i)}$ に関して，以下のようにシルエット係数を求める [?]．

＜凝縮度＞

$$a^{(i)} = \frac{1}{|C_{in} - 1|} \sum_{x^{(j)} \in C_{in}} \|x^{(i)} - x^{(j)}\| \tag{3.16}$$

＜乖離度＞

$$b^{(i)} = \frac{1}{|C_{near}|} \sum_{x^{(j)} \in C_{near}} \|x^{(i)} - x^{(j)}\| \tag{3.17}$$

＜シルエット係数＞

$$s^{(i)} = \frac{b^{(i)} - a^{(i)}}{max(a^{(i)}, b^{(i)})} \tag{3.18}$$

本研究では最小 2 つのクラスターから，最大 30 のクラスターまでのシルエット係数を計算し，それらの中で一番係数が高いクラスター数を採用する．

§ 3.3　単語間のつながりと共起語ネットワーク

関連性の高い単語は，一緒に出現することが多いため，それらの単語の共起関係を調べることで，単語間の関係性を理解することができる．共起分析では単語同士の Simpson 係数とい指標を用いて単語同士の共起度合いを比較し，共起関係にある単語と単語を線で結んで描かれる共起語ネットワークが利用される．このような共起語の分析を通じて，単語同士の意味的な特徴を理解することができる．本研究では，各クラスター内の単語にどのような関係があるのかを理解することを目的とする．共起関係を分析するには主に Jaccard 係数，Dice 係数，Simpson 係数が用いられる [?].

Jaccard 係数

ある集合 A とある集合 B について Jaccard 係数 $J\,(A,\,B)$ は，以下の式で定義される．

$$J(A,B) = \frac{|A \cap B|}{|A \cup B|} \tag{3.19}$$

Jaccard 係数は 2 つのデータセット間の類似度を測る手法である．2 つの集合に含まれる要素のうち共通要素が占める割合を表しており，完全に一致するときに 1，共通する要素がないときに 0 となり，係数は 0 から 1 の間の値となる．テキストマイニングにおいては，文章と文章の類似度を表す指標となる．具体的には 2 つの語少なくともどちらかが含まれる文章を数えて，2 つの語両方が含まれる文章の割合を計算する．割合が大きければ，2 つの語は今回のデータセットの中において「近い」と判断することができる．この Jaccard 係数が大きいほど 2 つの集合の類似度は高いといえる．

<Jaccard 係数の欠点 >

Jaccard 係数では分母に 2 つの集合の和集合を採用することで値を標準化し，他の集合同士の類似度に対する絶対評価を可能にしている．しかし，Jaccard 係数は 2 つの集合の差集合の要素数に大きく依存するため，差集合の要素数が多いほど Jaccard 係数は小さくなる．これは，人の目から判断した際の「共通要素が多いほど類似度が高い」という感覚と異なっている．

そこで，差集合の要素数の影響を抑え，共通要素の要素数の影響に重みをおく Dice 係数が提案された．

Dice 係数

ある集合 A とある集合 B について Dice 係数 $DSC\,(A,\,B)$ は，以下の式で定義される．

$$DSC(A,B) = \frac{2|A \cap B|}{|A| + |B|} \tag{3.20}$$

Dice 定数の定義式は，Jaccard 係数の定義式の分母 $|A| \cup |B|$ を $(|A| + |B|)/2$ と変換することで得られる．よって Dice 係数は 2 つの集合の平均要素数と共通要素数の割合を表しており，Jaccard 係数と同様に 0 から 1 の間の値となることがわかる．また，こちらも Jaccard

係数と同様に Dice 係数が大きいほど 2 つの集合の類似度は高いといえる．分母を「和集合の要素数」から「2 集合の平均要素数」とすることで，一方の集合だけ要素数が膨大である場合などに類似度が著しく下がる問題を防ぎ，共通要素数を重視した類似度を計算している．

<Dice 係数の欠点 >

上記でも説明した通り，Dice 係数の定義式は，Jaccard 係数の定義式の分母を「和集合の要素数」から「2 集合の平均要素数」とすることで，差集合の要素数が膨大になった場合に類似度への影響を緩和している．しかし，緩和しているとはいっても，2 集合の要素数に大きな差があり差集合の要素数が膨大になった場合 (例えば，一方の集合が別の集合を内包している等の場合) に，Dice 係数は低下してしまう．

そこで，差集合の要素数の影響を極限まで抑えた Simpson 係数が提案された．

Simpson 係数

ある集合 A とある集合 B について Simpson 係数 $overlap\,(A, B)$ は，以下の式で定義される．

$$overlap(A, B) = \frac{|A \cap B|}{min\{|A|, |B|\}} \tag{3.21}$$

上記の定義より，Simpson 係数は 2 つの集合のうち要素数が少ない方の要素数と共通要素数の割合を表しており，Jaccard 係数や Dice 係数と同様に 0 から 1 の間の値となることがわかる．また，Simpson 係数が大きいほど 2 つの集合の類似度は高い (よく似ている) といえる．Dice 係数の定期式は，Jaccard 係数の定義式の分母 $|A| \cup |B|$ を $(|A| + |B|)/2$ と変換することで得られた．これに対して Simpson 係数の定義式は，Dice 係数の定義式の分母を「2 集合の平均要素数」から「2 集合のうち少ない方の要素数」とすることで，Dice 係数よりも差集合の要素数による影響を下げ，相対的に共通要素数を重視した類似度計算を実現している．

<Simpson 係数の欠点 >

上記でも説明した通り，Simpson 係数の定義式は，Dice 係数の定義式の分母を「2 集合の平均要素数」から「2 集合のうち少ない方の要素数」とすることで，Dice 係数よりも差集合の要素数による影響を下げ，相対的に共通要素数を重視した類似度計算を実現している．しかし，Simpson 係数では要素数が少ない方の要素数を分母としているため，一方の集合の要素数が少ない場合に，差集合の要素数がどれだけ多くても類似度がほぼ 1 となってしまう．この問題を解決するためには，2 つの集合の要素数に条件 (閾値を設定する，2 集合間の要素数の差が範囲内である等) を付加するとよい．

共起語ネットワークを 3D グラフと 2D グラフによって可視化を行う．3D グラフと 2D グラフにはそれぞれメリットデメリットが存在する．

3D グラフのメリット，デメリット

- 単語の共起関係を 3 次元で表現できるため 2D グラフに比べて表現できる情報量が多い.
- 情報量の多さや 3 次元空間であることから，視認性が 2D グラフに比べて悪い

これらのことを踏まえ，3D グラフと 2D グラフの両方で共起語ネットワークを表示できるようにした．3D グラフの作成には Three.js，2D グラフの作成には pyvis を用いた．pyvis における Netowork でも 3D グラフを描画することができるが画面が固定になっており，ノードの数が増えると要素が絡み合って見づらいグラフになってしまう．そのため，3D グラフの描画には，3 次元での描画にたけている Three.js を用いる．

Three.js

Three.js はウェブブラウザ上で 3 次元コンピュータグラフィックスを描画すための JavaScript ライブラリである．HTML5 の規格に従っており，プラグイン不要で利用することができる．また，WebGL という 3D グラフィックス API をラッピングしており，簡素なコードで 3DCG を描画することができる．

3 次元コンピュータグラフィックスとは，3 次元の立体的な仮想物体を，コンピュータで演算することで平面上に奥行きや質感のある画像を表す手法である．従来は大型計算機が必要であったが，プロセッサの性能向上と GPU の一般化により，物性シミュレーションや 3D ゲームなど，さまざまな分野で利用されている [?].

描画には ”3D Force-Directed Graph ”というモジュールを用いており，Json ファイルでデータを与えることで，有向グラフを作成することができる．

pyvis

pyvis の Network は，Python の可視化ライブラリであり，ネットワークやグラフの作成と可視化に特化している．ネットワークグラフを作成した後 html ファイルとして出力することができ，出力された html には JavaScript が含まれているため自由な操作が可能である．このライブラリを用いることで，ノードやエッジを持つデータを簡単に可視化することができる．

Network は，インタラクティブなグラフ表示を行うことができ，ユーザーのマウス操作やタッチ操作によってグラフを探索したり，ノードやエッジの情報を表示したりすることができる．またノードやエッジの色やサイズなどの変更も容易であり，グラフの見た目の変更を自由に行える．さらに，NetworkX など別のライブラリで作成したグラフを pyvis で読み込んで出力することもできる [?].

共起分析による出力を可視化することで，単語同士のつながりを視覚的かつ直観的に理解することができるようになる．言葉での発想だけでなく，視覚的な情報や空間的な情報を用いて発想が行えると考える．共起分析の可視化によって情報の理解と分析を促進することを目的とする．

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

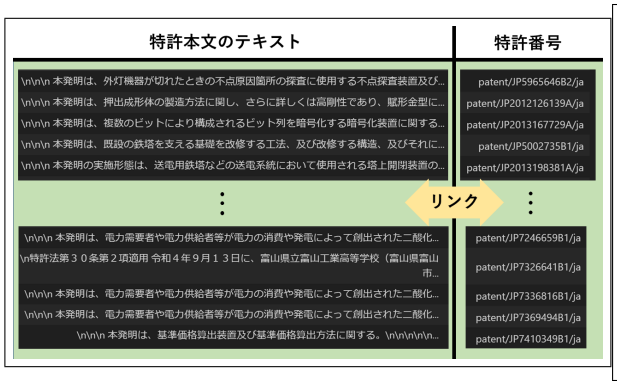


図 4.1: テキストデータのフォーマット

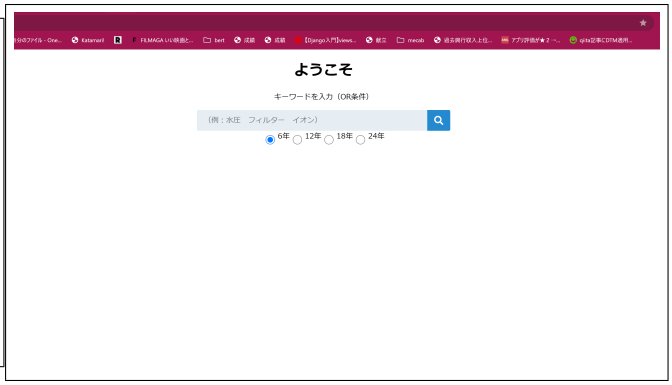


図 4.2: システムのフロントページ

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

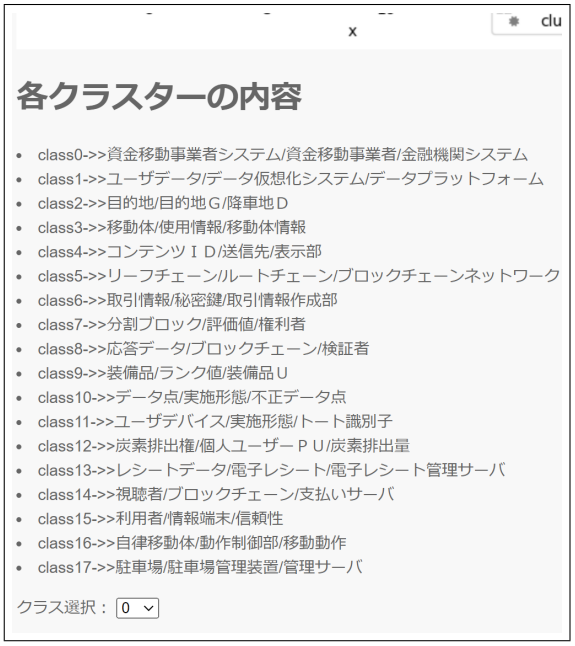
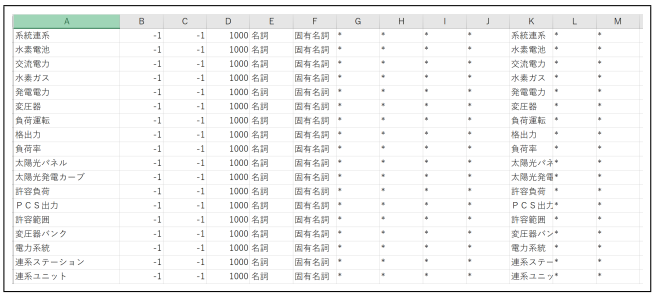


図 4.3: 解釈の出力例



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

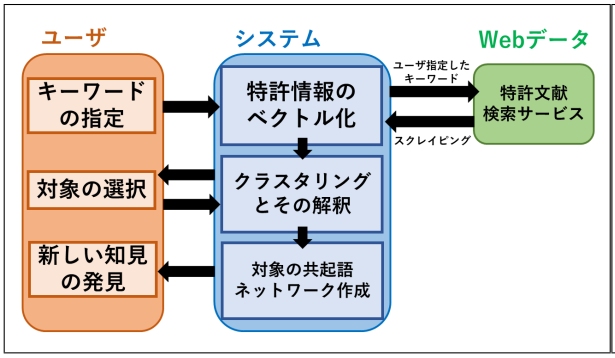


図 4.5: 提案システム

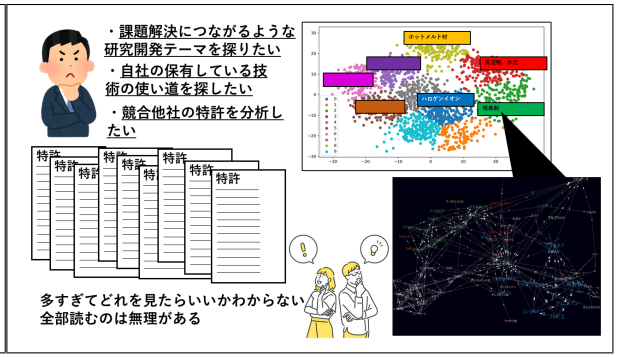


図 4.6: IPL への活用

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

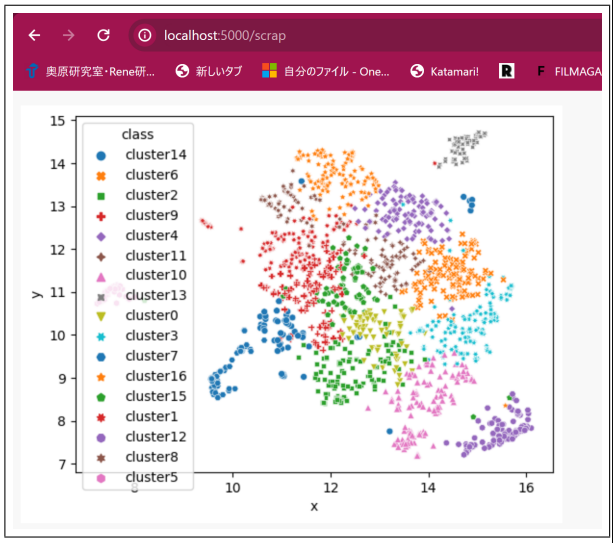


図 5.1: ベクトル化の結果

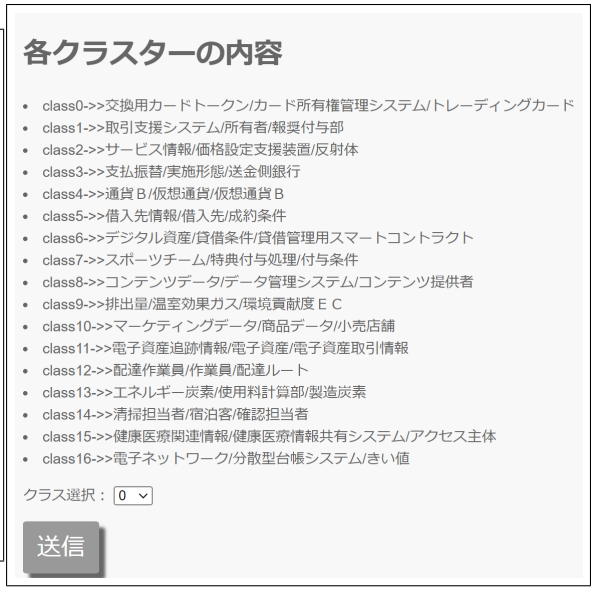


図 5.2: 出力されたタイトル

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

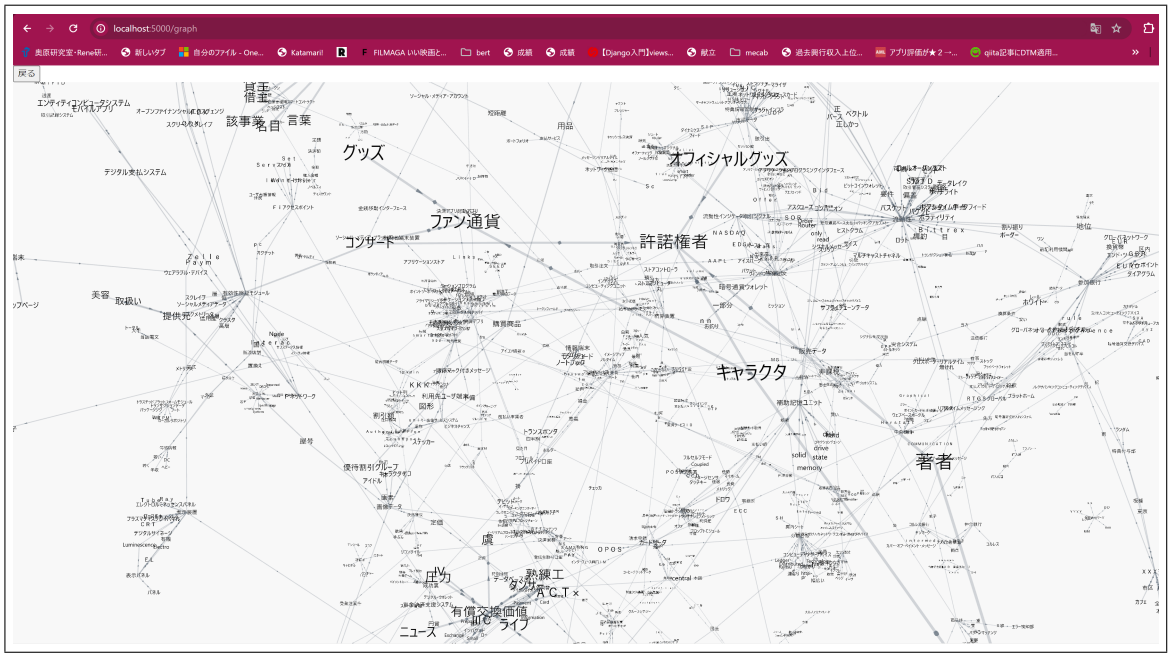


図 5.3: 出力された 3D グラフ

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] NEC ソリューションイノベータ, ”VUCA とは？意味や読み方、VUCA 時代の組織作りのポイントを解説”, 閲覧日 2024-02-04, https://www.nec-solutioninnovators.co.jp/sp/contents/column/20230623_vuca.html.

[2] 株式会社三菱総合研究所, ”代 4 次産業革命における産業構造分析と IoT・AI 等の発展に係る現状及び課題解決に関する調査研究”, 閲覧日 2024-02-04, https://www.soumu.go.jp/johotsusintokei/linkdata/h29_03_houkoku.pdf.

[3] 特許庁, ”広報誌「とっきょ」”, 閲覧日 2024-02-04, https://www.jpo.go.jp/news/koho/kohoshi/.

[4] WPIO, ”世界知的財産指標報告書”, 閲覧日 2024-02-04, https://www.wipo.int/pressroom/ja/articles/2023/article_0013.html.

[5] 山元 悠貴．”Web 内容マイニングによる複数キーワードに対する 3D 有向グラフを用いた発想支援”．富山県立大学学位論文 2020.

[6] 特許庁, ”経営戦略に資する知財情報分析・活用に関する調査報告書”, 閲覧日 2024-02-04, https://www.jpo.go.jp/support/general/document/chizaijobobunseki-report/chizai-jobobunseki-report.pdf.

[7] 東京知的財産総合センター, ”中小企業経営者のための知的財産戦略マニュアル”, 閲覧日 2024-02-04, https://www.tokyo-kosha.or.jp/chizai/manual/senryaku/rmepal000001vypy-att/senryaku_all_vol.9.pdf.

[8] 特許庁, ”経営戦略を成功に導く知財戦略”, 閲覧日 2024-02-04, https://www.jpo.go.jp/support/example/document/chizai_senryaku_2020/all.pdf.

[9] 特許庁, ”「経営戦略に資する知財情報分析・活用に関する調査研究」について”, 閲覧日 2024-02-04, https://www.jpo.go.jp/support/general/chizai-jobobunseki-report.html.

[10] 高橋 成夫，”経営戦略論の一動向について”，新潟産業大学経済学部紀要．2019．53 号，pp．7-17.

[11] 金融ナビ, ”経営戦略の策定に役立つフレームワーク 7 つ｜経営戦略の代表例も解説”, 閲覧日 2024-02-04, https://financenavi.jp/basic-knowledge/management_strategy_framework/#tag1.

[12] 特許庁, ”2019 年度_知的財産権制度入門”, 閲覧日 2024-02-04, https://www.jpo.go.jp/news/shinchaku/event/seminer/text/document/2019_syosinsya/1_3.pdf.

[13] 正林国際特許商標事務所, ”既存技術をほかの用途へ転用する，あるいはビジネス上の課題を解決する既存技術を模索するためのIP ランドスケープの活用”, 閲覧日 2024-02-04, https://www.wipo.int/edocs/plrdocs/en/plr_2019_shobayashi_other.pdf.

[14] Acrovision, ”自然言語処理とは？”, 閲覧日 2024-02-04, https://www.acrovision.jp/career/?p=2820.

[15] 株式会社　日立ソリューションズ・クリエイト, ”テキストマイニングとは？ 手法や活用法を解説”, 閲覧日 2024-02-04, https://www.hitachi-solutions-create.co.jp/column/technology/text-mining.html.

[16] gikyo.jp, ”Perl による自然言語処理入門”, 閲覧日 2024-02-04, https://gihyo.jp/dev/serial/01/perl-hackers-hub/0031011.

[17] AGIRobots Blog, ”【Transformer の基礎】Multi-Head Attention の仕組み”, 閲覧日 2024-02-04, https://developers.agirobots.com/jp/multi-head-attention/.

[18] Nils Reimers，Iryna Gurevych．”Sentence-BERT：Sentence Embedding using Siamese BERT-Networks”, *ArXiv e-prints*, 1908. 10084, 2019

[19] data-analytics.fun,”【論文解説】Sentence-BERT を理解する”, 閲覧日 2024-02-04, https://data-analytics.fun/2020/08/04/understanding-sentence-bert/.

[20] Mclnnes，L.， Healy，J.，Melville，J. ”UMAP：Uniform Manifold Approximation and Projection for Dimension Reduction”, *ArXiv e-prints*, 1802. 03426, 2018

[21] Hatena Blog, ”UMAP の仕組み–低次元化の理屈を理解してみる”, 閲覧日 2024-02-04, https://kntty.hateblo.jp/entry/2020/12/14/070022.

[22] 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.

[23]

[24] 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.