修士論文

単語トピック特定性を考慮した単語ベクトルの重み付けに関する研究

1450021　中山雄貴

主指導教員　Ho Tu Bao

審査委員　Ho Tu Bao (主査)
Dam Hieu Chi
西本一志
林幸雄

北陸先端科学技術大学院大学
知識科学研究科

提出年月:平成 28 年 2 月

目次

図目次

表目次

第 1 章 序論

1.1　背景と目的

現在、インターネットや記憶端末の発達によって、電子化された文書の量が爆発的に増えてきた。このような大量に電子化された文書を我々は最大限に活用する必要がある。そういった言語資源を最大限に活用するには、目的別に整理したり、要約したりする必要があるが、それらを人手で行うことはほぼ不可能である。その結果、コンピュータを使って自動的にそれらのタスクを行うことが、計算機科学やデータ科学における主たる研究対象の一つとなっている。また、自然言語を理解し、それらを巧みに扱うことを目的として発展した計算機科学の分野のことを自然言語処理という。自然言語処理のタスクをより正確に行うためには、文書の表面上の情報を分析するだけではなく、潜在的な関係情報を計算処理可能な形式で表現し、うまくモデルに組み込む必要がある。単語の意味に関しても例外ではなく、自然言語処理のタスクをより精密に行うためには、単語同士の意味的な関係性を言語モデル内に組み込み、コンピュータに理解させることが非常に重要な課題となっている。例えば、文書分類のタスクにおいてコンピュータに”発動機”と”エンジン”に関連性があるかどうかを理解させることができなければ、たとえ 2 つの文書の内容が類似していても、執筆者のくせにより片方には”エンジン”という単語しか現れずに、もう一方には”発動機”という単語しか現れなかった場合、それぞれの文書を同じカテゴリに分類させることは困難である。一方、コンピュータに単語の意味の関係性を理解させることができれば、文書分類の精度をより良くするだけでなく、ある人が予期することができなかった関連した文書を同じカテゴリに分類することで、その人を新しい発想へ転換させる手助けをすることができるかもしれない。しかしながら、人間が複数の概念や単語同士の意味の関係性を考慮しながら文書を理解することを容易に行

えているにも関わらず、自然言語処理において、コンピュータに単語あるいは文脈の意味的な関係性を認知させることは非常に難しく、まだ道半ばである。

自然言語処理の研究において、単語同士の意味的な関係をコンピュータ上で表現する際、多くの言語学者たちはシソーラスやオントロジーといった特定分野の専門家よる手作業で構築された言語資源を使用してきた。シソーラスやオントロジーは用語の様々な関係を階層構造などによって構造的に表現した語彙集合のことであり、様々な自然言語処理のタスクにおいて、コンピュータに意味の概念を取り込む際、非常に使い勝手が良かったからである。しかし、そのような言語資源を設計するためには、非常にコストや時間がかかるという問題点がある。さらに近年、医療分野などの様々な専門分野において、新語あるいは造語が次々と登場している。そのため、すべての語句の意味的な関係性を、言語資源を設計する注釈者が理解し、構造化するのは年々困難になってきている。そういった状況の中で、そのような言語資源を人間ではなく、コンピュータが自動的に構築する技術の重要性が唱えられてきている。

人がどのように単語の意味を理解しているのかについて、認知科学の分野などで研究が盛んに行われているが詳細についてはいまだに未知である。しかし、類似した意味を持つ単語同士の文書における振る舞いは半世紀前から解明されてきている。その代表例が分布的仮説である。分布的仮説に基づいたアプローチでシソーラスやオントロジーのように単語と単語にどういった性質の意味的関係(IS 関係など)があるのかといった情報を知ることはできないが、単語と単語の関係の強さを量的推計によって表現できる。20 年以上前にコンピュータによって、電子文書の分布的統計から単語間の意味的な類似度を自動的に測定できるようになり、現在まで分布的仮説に基づいたアプローチによる研究が盛んに行われてきた[4]。

分布的仮説とは、言語学者の Firth や Harris の研究によって形成された文章中の単語の意味に関する仮説である。ある単語とその他の単語の意味的な関係はそれらの単語の文脈においての出現パターンの類似性によって推定できるという仮説である。単語空間モデル(Word Space Models, WSM)とは、前述した分布的仮説の前提をもとに、ある単語とその前後 N 単語内に出現する単語との共起頻度を求め、それらをベクトルに記録し、文脈(一般的には共起単語)を次元とした高次元空間においてのベクトルとして単語の意味を表現するモデルである。また、前述の方法から求められた各単語のベクトルの距離の近さをコサイン類似度やユーグリット距離など用いて計算することにより、前後 N 単語に出現する単語の分布の類似性を計算することで、それらのベクトルを持つそれぞれの単語がどの程度類似した意味を持っているのかを表現することができる[1]。WSM は単語レベルの意味を表現するのに単純だがとても効果的な手法である

という実用的な理由から、単語曖昧性解消、クエリ拡大など、様々な自然言語処理分野において大きな影響を与えた。

WSM における研究は、計算効率性、ベクトルの類似度尺度、文脈の選択など様々であるが、本研究においては、単語ベクトルの重み付けに焦点を当てる。単語の意味ベクトルを生成する際、単語の共起頻度から生成されたベクトルは、出現頻度の高い文脈パターンを必要以上に重視してしまうために単語の意味の比較にそれらのベクトルをそのまま用いてしまうと、精度が非常に悪くなってしまい、適切な類義語を抽出することができなくなる。そこで、多くの WSM において用いられる手法は重み付けである。重み付けとは、ある要素にだけしか現れない事例により大きな重みを与え、どの要素にもみられる事例にはより小さな重みを与えることである。特定の要素にしか起こらない事例は、単語ベクトルにおける共起単語に関する特徴を豊かにし、ベクトル同士を比較する際、非常に重要な要素となるからである。例えば、「猫」という単語ベクトルがあったとしよう。単語ベクトル同士を比較する際、「鎖」や「ペット」という共起単語は単語ベクトルにとって重要な特徴となるが、「好き」や「持つ」といった単語は、「猫」という単語以外にも様々な単語と共起するため、あまり重要な特徴ではない。よって、そのベクトルにおいては、「鎖」や「ペット」といった単語により大きな重みを与えるのが重み付けである。このような重み付け手法は様々なものが提案されている。その中でも特に有名なものとして、PMI (Point-wise Mutual Information)と t 検定がある。PMI や t 検定はある単語ともう一方の単語が共起する期待頻度に比べてどれぐらい実際の共起頻度が高いかを推定する評価指標である。大概の場合において PMI や t 検定は良い重み付けを行うが、「最近」や「いつも」といったどの単語にとっても一般的で単語の意味比較に役立たない単語であっても、それらの単語との共起頻度が期待頻度を大きく超えていれば、より大きな重み付けをしてしまうという問題点がある。単語の意味は、様々な側面から決定されるにも関わらず、PMI や t 検定は共起性という 1 つの側面しか考慮することができないのである。PMI や t 検定以外の従来の重み付け手法においても共起頻度の期待値との比較によって重み付けを行う場合がほとんどであり、共起性依存の問題を持つ手法がほとんどである。

したがって、本研究の目的は、単語ベクトルの重み付けの共起性依存の問題を軽減することによって、重み付けの質を向上させ、単語ベクトルの弁別性を改善させることとする。我々は重み付けの共起性依存の問題を解決するために単語そのものの性質に着目することにした。文脈単語自身の具体性を単語ベクトルの重み付けを行う上で重要なもう一つの側面とした。その具体性の概念を定義できるようにするため、具体的な単語であればあるほど、特定の分野の文脈のみで出現すると仮定し、抽象的な単語であればあるほど、様々な分野の文脈におい

て出現すると仮定した。そして、その分野を表現するために潜在 Dirichlet 配分法を用いて同じ文書で出現しやすい語彙集合を表現するトピックとし、単語におけるトピックの確率分布と最も抽象的な単語のトピックの確率分布であると仮定した、トピックの確率分布の違いを求めることによって、単語トピック特定性という単語がどれくらい特定のトピックに出現しているのかを示す指標を定義した。この指標を定義することにより、ある共起単語がどのくらい具体的な意味を持っているかを判断し、相対的に大きな重みを具体的な単語に加えられるようにする。この単語がどれくらい具体性を持っているかを示す単語トピック特定性と共起に基づく重みを組み合わせたうえで、重み付けを行うことにより、重み付けの共起性依存の問題を解決し、より弁別性がある単語の意味ベクトルを生成することを目指す。さらに、単語トピック特定性を、言語資源を用いることなく、教師なしで定義することも目標とする。

1.2　論文の構成

この節では、この論文の構成を説明する。まず、第 2 章においては単語空間モデルの概要について説明した。第 3 章では、単語空間モデルにおいて用いられてきた既存の重み付けの手法について、その性質に着目しながら、その問題点を明らかにした。第 4 章では、第 3 章で明らかになった問題を解決するための方針を示し、その方針に沿った新しい重み付け手法を提案する。潜在 Dirichlet 配分法と Jensen-Shanon ダイバージェンスを用いて、単語トピック特定性を計算し、その単語トピック特定性と PPMI や t 検定などの共起に基づく重み付け手法を組み合わせて各単語ベクトルの成分に重み付けを行う手法を提案する。第 5 章では、新しく提案した手法が、実際に効果を発揮するのか確証を得るために行った Wikipedia のデータを用いた実験の結果を紹介した。そして第 6 章では、その実験の結果に対する議論を行い、第 7 章では、本研究を総括した。

第２章 単語空間モデル

2.1　分布的仮説

自動的に文書を解析する上で、最も重要なタスクの一つとして単語同士の意味的な関係の推定がある。その中でも分布的仮説に基づいたアプローチは、多くの研究者によって研究されてきた。分布的仮説とは、類似した文脈上で出現する単語は意味的に類似しているという前提のことをいう。分布的仮説は言語学者の Harris と Firth の研究を発端としている。Harris は単語の現れる文脈を観察することによって、その単語の言語的な役割を知ることができると論じており[5]、Firth は単語の特徴的な単語の配置から単語の意味を決定できると論じた[6]。これらの研究から後に単語の意味はその周りに存在する単語の分布によって推定することができるという分布的仮説が生まれることになる。分布的仮説は普段から人間が、知らない単語の意味を推定するために使っている言葉に関する性質の一つでもある。例えば”Bing”という単語の意味を以下の例文から推測できるだろう。

Bing を使えば簡単に探しているものを見つけられる。
Bing の検索結果をみてみた。
Bing はあまり検索の精度が良くない。
Bing に新機能が導入された。
Bing に対応するプログラムをインストールしたい。
Bing の使い方をインターネットで調べる。
Bing には DiscoverBing と呼ばれるプロモーションサイトがある。
Bing は Google と似たようなものである。

日本ではあまりなじみがないが、実際”Bing”は世界的に有名な検索エンジンである。上記の例文を見れば、”Bing”が検索エンジンのようなものであるというこ

とを推測できる。我々はどのようにしてその意味を推測したのだろうか。周囲の単語から推測したのだろう。上記の例文において”Bing”以外で出現する単語には”検索”、”精度”、”機能”などがある。このように周囲の単語を観察していけば、単語がどのような意味を持っているのかを正確に推測することができる。このような単語を推定する際、人が自然に用いている、類似した単語は類似した文脈上で出現するという性質を分布的仮説と呼ぶ。この分布的仮説が単語の意味的な関係を得るために単語同士の共起情報を用いることに対する正当な理由を与えてくれる。この分布的仮説に基づいて、単語の出現する文脈をパターンとして抽出し、そのパターンの類似性を計算することによって類似した単語がどの単語であるかを判別することができるようになる。このような単語の意味を高次元の文脈ベクトルによって表現する手法のことを単語空間モデルという。次節では、単語空間モデルについて説明していく。

2.2　単語空間モデル

単語空間モデルは自然言語処理の研究者によって数十年間にわたって研究されてきた。単語空間モデルは、一般的に列 F_w が目標単語 w を表現し、それぞれの行 F_c が文脈 c を表現する共起行列 F におけるデータを集めることによって生成される[7]。最も典型的な例では、文脈は単語が出現する文書や単語に対する共起単語によって表現される。単語空間モデルにおいて、前者の場合は単語文書行列として、後者の場合を単語単語行列として各単語の特徴を表現する。

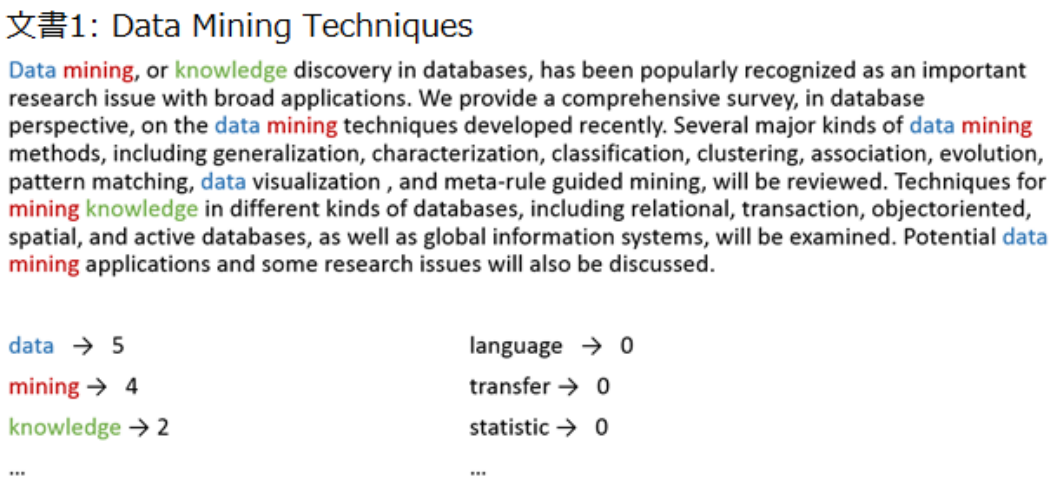

図　2.1 単語文書行列の成分の記録方法

	文書1	文書2	文書3	文書4	文書5
単語1	101	11	39	44	75
単語2	150	9	83	64	40
単語3	121	59	37	88	54
単語4	107	103	11	109	30
単語5	110	35	25	54	50
単語6	131	94	61	11	149
単語7	108	82	65	21	112
単語8	116	17	149	85	37

図 2.2 単語文書行列の例

文脈を単語が出現する文書として表現する場合、それぞれの文書 d に目標単語 w が何回出現したかを単語文書行列の要素 F_{wd} に記録していき、その要素を単語の文脈についての分布的な情報とする。図 2.1 における文書 1 において、目標単語を"data"とするとその文書内で出現した頻度は 5 であるので、$F_{"data",1}=1$ と記録する。そしてすべての単語に対する文書頻度を観察していくと、図 2.2 のような行列を生成することができる。なお、単語文書行列において、bag-of-words (BOW)表現、つまり単語の出てきた順番は考慮しない。

しかし、このように各単語に対する各文書においての出現頻度を記録していく単語-文書行列の方法には少し問題がある。文書を文脈とするために共起情報が非常に粗くなっているのである。1 つの文書といえども、文脈は変化する。例えば、図 2.1 の文書において、"knowledge"を含まない文がある。さらに、"knowledge"を含む 2 文において共通している"mining"や"database"といった単語もあれば、片方の文にしか出現しない単語もある。このような文脈の変化を、文脈を文書とする方法では感知することができない。ここで、考えられるこの問題に対する対処策は、文脈を文書単位からどんどん短くしていくことである。文脈を短くすれば、文章における細かな文脈の変化に対応でき、文脈の変化をより細かくとらえたパターンを抽出することができる。より短い文脈においての共起パターンを得るために、ここである単語の前後 N 単語から成る文脈ウィンドウという概念を導入する。

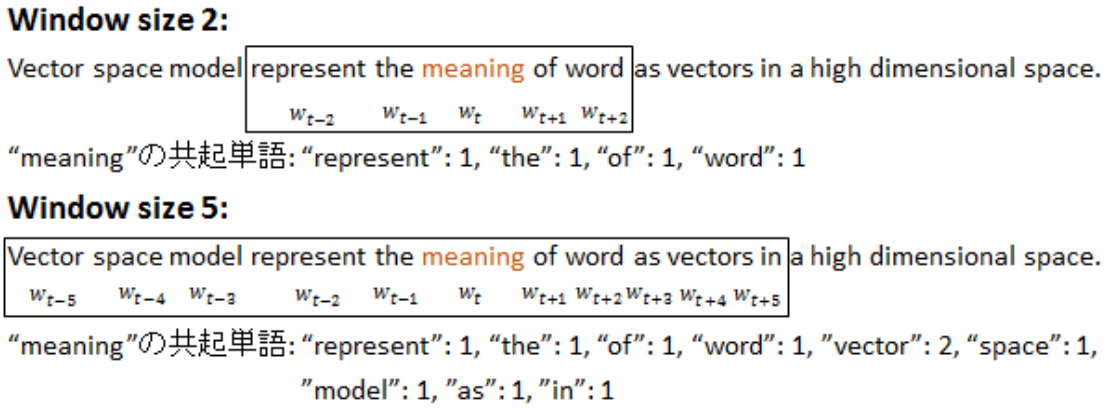

図　2.3 文脈ウィンドウの説明

文脈ウィンドウのように文脈を短くした単語単位は文脈の表記において単語文書行列とは異なる見方が必要となることを説明する。単語文書行列を生成する際、同一文書において、出現する単語の種類は豊富にあり、それらの出現頻度も多い。また、文書数は単語の種類と比較して少ない。しかし、数単語から形成される文脈ウィンドウで単語文書行列と同じ登録方法を採用してしまうと、文脈において出現する単語はほとんどないため、非常にスパースになる[3]。さらに、文書数に比べてウィンドウ数は非常に多いため、非常に高次元のベクトル空間となってしまうため、ベクトル同士を比較することが困難となる。そこで、文脈ウィンドウを使う際は、ある単位の中に現れる単語の頻度を記録するのではなく、文脈単位を構成する目標単語以外の単語を文脈としてみなし、目標単語と文脈単位内において共起する単語を文脈の分布的なパターンとする。文脈ウィンドウを文脈とする場合、それぞれの文脈(単語)c と目標単語 w が文脈ウィンドウ内で共起した頻度を単語単語行列の要素 F_{wc} に記録していく[7]。一般的には文書を文脈単位とする方法同様に単語の順番は考慮せず、BOW 文脈ベクトルとして文脈が表現される。

	単語1	単語2	単語3	単語4	単語5
単語1	13	34	78	52	35
単語2	34	1	78	33	76
単語3	78	78	16	86	14
単語4	52	33	86	4	90
単語5	35	76	14	90	19

図　2.4 単語単語行列の例

以上のように単語文書行列または単語単語行列の頻度による行列を生成することができるのだが、頻度行列には問題がある。高頻出語はどの単語ベクトルにおいても大きい重みを持っているために、単語の文脈に関する特徴を削いでしまい、単語ベクトル同士の類似度を計算する際、その精度を悪くしてしまうのである。例えば、”also”、”now”、”much”というような単語と共起していても、それらの単語はほとんどの単語とたびたび共起するため、単語ベクトルの特徴を与えるために役立つとは考えらない。逆に考慮してしまうことで、単語ベクトルの特徴を削いでしまう。そこで、一般的には 2 つの単語がほかの単語に比べてどれくらい共起しやすいかなどを考慮することによってその共起に対する重要度を評価し、単語の意味比較をより精密に行うために単語ベクトルの各要素への重み付けが行われる[4]。次節では重み付けについて説明する。

2.3　単語ベクトルの重み付け

単語空間モデルにおいて、重み付けとは相対的な意味関係を知覚する際に文脈単語が目標単語に対してどれくらい重要度を持つのかを各要素の重みを変化させることによって表現することをいう。例えば、医療用語に対する意味の比較を行いたい場合はより一般的な”history”や”building”のような文脈単語には小さな重みを与え、”cardiac”や”artery”など症状や病名に関連した文脈単語にはより大きな重みを与える。単語空間モデルの性能は単語ベクトルの要素を決定する文脈単語の重み付けに依存しているといっても過言ではなく、適切な文脈単語への重み付けを行うことによって、単語空間モデルから生成されたベクトルの類似度によって意味的関係性をみるタスクの性能を大幅に改善することができる。昨今の研究の中で最も頻繁に用いられるアプローチは目標単語のベクトルの各文脈単語に記録されている共起頻度の評価値を目標単語と文脈単語との共起の起こりやすさに応じて変化させることである。多くの単語にとって頻繁に起こり得る出来事に対してはより小さな重みを与え、あまり起こりえない出来事に対してはより大きな重みを与える。前述の例の”Bing”の単語の意味を知るうえで、”search”、“system”と”see”、”use”という 2 つの単語群があった場合、たとえ、後者の単語群が頻繁に共起していても前者の単語群は後者の単語群に比べて重要になる。

このアプローチに基づいた単語ベクトルにおいて使われる最も有名な重み付け法として Point-wise Mutual Information(PMI)がある。PMI は、x と y が、それぞれを独立として仮定したときの期待度に比べて、どれくらい頻繁に起こるのかをみる尺度である[2]。PMI は以下のように定義される。

$$\mathrm{PMI}(\mathrm{x},\mathrm{y}) = \log\frac{p(x,y)}{p(x)p(y)} \quad (2.1)$$

p(x,y)は x と y が同時に出現する確率を示していて、x を目標単語、y を文脈単語とする場合、p(x,y)=c(x,y)/N となる。c(x,y)は x と y が共起する頻度であり N はコーパスにおいて出現する単語の数である。PMI は正から負までの値を取る. しかしながら、負の PMI は x と y が、それぞれを独立として仮定したときの期待度に比べて、あまり頻繁でないことを示しているのだがその値は、理論的に予測することが難しく信頼できないため、単語ベクトルの形成において不利に働く。これゆえ、一般的には負の PMI をすべてゼロに置き換える。Positive Point-wise Mutual Information (PPMI)を用いる。単語の意味ベクトルを用いて単語の意味の比較を行う際、PPMI で重み付けを行うほうが、PMI で行うよりもたいていの場合はより良い性能を示すことが Bullinaria と Levy の研究[8]によって確認された。PMI を用いて単語-単語頻度行列の重み付けを行うと最終的には図 2.4 のような行列が得られる。

	単語1	単語2	単語3	単語4	単語5
単語1	−0.45774	−0.06022	0.21218	0.04741	−0.07050
単語2	−0.06022	−1.61172	0.19216	−0.17010	0.24623
単語3	0.21218	0.19216	−0.58403	0.15767	−0.57667
単語4	0.04741	−0.17010	0.15767	−1.16344	0.24277
単語5	−0.07050	0.24623	−0.57667	0.24277	−0.37869

図 2.5 重み付き行列の例

以上の手順から単語の意味に関するベクトルが生成できる。

2.4 類似度尺度

前節までは、ベクトルを生成する方法についてみてきた。ここからは、その生成した単語ベクトルをどのように比較するのかについて記述していく。

ベクトル同士を比較する際には一般的には距離尺度を使う。Euclidean 距離、Manhattan 距離や情報理論においての距離である Hellinger 距離、Bhattacharya 距離、Kullback-Leibler 距離など様々な距離が提案されてきたが、Bullinaria と Levy の研究[8]によると、最善の距離尺度はコサイン距離である。そして、たいていの場合においても、意味ベクトル同士を比較する際はコサイン距離を使用する。文脈単語を n 単語持つ単語 x と y の意味ベクトルを次のよう

におく[4]。

$$\mathbf{x} = (x_1, x_2, \dots, x_n)$$
$$\mathbf{y} = (y_1, y_2, \dots, y_n)$$

するとベクトル同士の内積は

$$\mathbf{x} \cdot \mathbf{y} = \sum_{i=1}^{n} x_i y_i = x_1 y_1 + x_2 y_2 + \cdots + x_n y_n$$

のように定義され、ベクトルの長さは

$$|x| = \sqrt{\sum_{i=1}^{n} x_i^2}$$

のように定義される。x と y の間の角度は以下のように計算される。

$$\cos(\mathbf{x}, \mathbf{y}) = \frac{\boldsymbol{x} \cdot \boldsymbol{y}}{|\boldsymbol{x}||\boldsymbol{y}|} = \frac{\sum_{i=1}^{n} x_i y_i}{\sqrt{\sum_{i=1}^{n} x_i^2} \sqrt{\sum_{i=1}^{n} y_i^2}} \quad (2.2)$$

コサイン角度が-1 になれば x と y は全く逆の意味を持った単語同士となり、1 に近くならばなるほど x と y は類似した意味を持っていることになる。PPMI で重み付けをした場合は、0 に近くなれば、真逆の意味を持ち、1 に近くなれば同義の意味を持つことになる。例えば、図 2.4 の単語ベクトルにおいて単語 1 と単語 2 の類似度はベクトルがそれぞれ以下のようになる。

$$\boldsymbol{w}_1 = (-0.45774, -0.06022, 0.21218, 0.04741, -0.07050)$$
$$\boldsymbol{w}_2 = (-0.06022, -1.61172, 0.19216, -0.17010, 0.24623)$$

よって $\cos(\boldsymbol{w}_1, \boldsymbol{w}_2) = 0.\ 30884$となる。

第３章 関連研究

本章では単語ベクトルの重み付け手法について説明する。それぞれ手法にはどのような特徴があるのかを述べた後、それらの手法に共通する問題点について説明する。

3.1　重み付け

単語空間モデルにおいて最も重要な要素は共起単語による文脈の表現である。単語空間モデルにおいて、単語の意味ベクトルの要素は共起単語との共起頻度を登録することによって生成される。しかし、共起頻度をそのまま用いてしまうと文脈単語の頻度による影響を受けるということを前章において説明した。共起頻度をそのまま用いてしまうことによって、機能語や高頻度語により大きな重みを与えてしまうために単語の意味ベクトルを比較する際、すべての単語に共通する共起単語の頻度ばかりの類似性を重視されてしまうため、単語ベクトル同士の類似度比較の精度が低下する。そこで単語の比較に対してより良い単語ベクトルにおいての文脈の表現をできるように単語の意味比較の際に重視すべき文脈単語の重み付けを行う。以下では、文脈単語の重み付け手法について説明する。

まず、文脈単語の条件確率を用いる手法である。目標単語に対する文脈単語の重みを文脈単語の条件確率によって定義すると以下のようになる。

$$\mathrm{weight(x, y)} = \frac{f_{xy}}{f_y} = p(x|y) \quad (3.1)$$

ここでf_{xy}は単語 x と単語 y の共起頻度であり、f_yは単語 y の出現頻度である。上記の尺度によって各単語ベクトルの要素の重み付けを行うと、共起頻度をそ

のまま用いる場合と比較して、より頻繁に出現する単語の影響を緩和させることができる。機能語などの高頻度語は、単語のベクトルを特徴づけるうえであまり重視するべきではないが、上式によって、単語の意味ベクトルの各要素は共起単語の出現頻度に対する目標単語と文脈単語の相対的な頻度を重みとなるので、たとえ、非常に大きな頻度を持っている文脈単語との共起頻度がほかの文脈単語との共起頻度と比較して非常に大きな場合であっても、その文脈単語の出現頻度が共起頻度を軽減するためにより小さな値となる。例えば、2 つの頻度による単語ベクトルが、それぞれ$w_1 = (15, 90, 3)$、$w_2 = (0, 80, 35)$でそれぞれの共起単語の頻度が 50、200、40 あった場合、そのまま、コサイン類似度を計算した場合、0.916 となるが、上式によってベクトルの要素を頻度から条件確率に変換すると、単語ベクトルはそれぞれ$w_1 = (0.3, 0.45, 0.075)$、$w_2 = (0, 0.4, 0.875)$となり、コサイン類似度は 0.468 となる。つまり、この重み付け手法を用いることによって高頻出語を必要以上に重視しないようにできる。

前述したような手法は、ただ単に高頻度語には小さな重みを与えて、低頻度語には大きな重みを与えるだけである。重み付けをより洗練された方法で行うためには、特異的な事例により大きな重みを与え、一般的な事例に対してはペナルティを与えることである。言い換えれば、それぞれの単語が独立に出現すると仮定したときに期待される共起頻度に比べて、実際の単語同士の共起がどれくらい頻繁なのかを計算することである。この考え方に基づいた重み付け手法には、最も有名なものとして、前章で説明した Pointwise Mutual Information (PMI) と t 検定がある。

まず、PMI について説明する。N をコーパスにおける単語のすべての数、f_{xy}を単語 x と単語 y の共起頻度であり、f_xを単語 x の出現頻度、f_yは単語 y の出現頻度とすると PMI は以下のように定義される。

$$\mathrm{PMI}(\mathrm{x}, \mathrm{y}) = \log \frac{f_{xy} N}{f_x f_y} \quad (3.2)$$

ここで log は自然対数とする。また$p(x) = f_x/N$、$p(y) = f_y/N$、$p(x, y) = f_{xy}/N$と置くと PMI は以下のように変形できる。

$$\mathrm{PMI}(\mathrm{x}, \mathrm{y}) = \log \frac{p(x, y)}{p(x)p(y)} \quad (3.3)$$

上式において、単語同士の偶然の共起はあまり重視されない。また、前章でも述べたように負の PMI は x と y が、それぞれを独立として仮定したときの期待度に比べて、あまり頻繁でないことを示しているがその値は、信頼できないため、一般的には負の PMI をすべてゼロに置き換える Positive Pointwise Mutual

Information(PPMI)が使われる[1]。

$$\mathrm{PPMI(x,y)} = \begin{cases} PMI(x,y)\ if\ PMI(x,y) > 0 \\ 0 \quad otherwise \end{cases} \quad (3.4)$$

上記の PMI による手法では低頻度の事例を重視してしまうという問題点がある。そのため下記のような positive localized mutual information (PLMI)[9]、positive normalized pointwise mutual information (PNPMI)[10]のような手法が提案されている。LMI は、相互情報量に目標単語の出現確率を掛けることによって PMI の低頻度を重視する問題を解決する。また、NPMI は、PMI の最大値が 1 になるように正規化することによって、低頻度バイアスを除去する。

$$\mathrm{PLMI(x,y)} = \begin{cases} LMI(x,y)\ if\ PMI(x,y) > 0 \\ 0 \quad otherwise \end{cases} \quad (3.5)$$

$$\text{where } \mathrm{LMI(x,y)} = p(\mathrm{x,y})\log\frac{p(x,y)}{p(x)p(y)}$$

$$\mathrm{PNPMI(x,y)} = \begin{cases} NPMI(x,y)\ if\ PMI(x,y) > 0 \\ 0 \quad otherwise \end{cases} \quad (3.6)$$

$$\text{where } \mathrm{NPMI(x,y)} = \frac{PMI(x,y)}{-\log(p(y))}$$

前述したアプローチに基づく手法で、もう一つの有名な手法に t 検定がある。PMI は機能語などのような高頻出単語を取り除く能力は優れているのだが、低頻度単語に対しては上手く働かないという問題点があった。つまり、低頻度の共起情報を重視しすぎてしまう傾向があった。PLMI や PNPMI といった手法を用いることで、低頻度単語を重視しないようにすることができるが効果は限定的である。t 検定は単語自身の頻度を考慮することによって、そういった問題を解決する。t 検定は PMI のように共起の強さを測るのではなく、共起があると言い切れる信頼性、つまり有意性を評価する[11]。一般的には t 検定は以下のように分散によって標準化された観測された平均と期待される平均の違いを計算する。

$$\mathrm{t} = \frac{\bar{x} - \mu}{\sqrt{\frac{s^2}{N}}} \quad (3.7)$$

t が高ければ、観測された平均と期待された平均が等価であるという帰無仮説を

棄却することによってより大きな尤度を与える。単語同士の共起の有意性を比較する際、この帰無仮説を 2 つの単語は独立であるという仮説、つまり p(x,y)=p(x)p(y) であるという仮説に置き換える。すると t 検定は、分散 s^2 が期待確率 p(x)p(y)に近似され、N を無視することによって以下のようになる[4]。

$$\mathrm{T^{-}test(x,y)} = \frac{p(x,y) - p(x)p(y)}{\sqrt{p(x)p(y)}} \quad (3.8)$$

以上が、重み付けの既存の手法の一般的な例である。上記において説明した PMI や t 検定などの手法は共起情報を上手く使うことによって、驚くべき共起事例に対してより大きな重みを当て、すべての単語に対して一般的な事例に対してはより小さな重みを与えるという、単語ベクトルに対して、正当な重み付けを行っている。しかし、共起性を重視しすぎるあまり共起した単語はどういったものなのかというような単語そのものにつての情報を考慮しないとい問題もある。その問題点については次節において詳しく説明したい。

3.2 共起頻度を用いる重み付けの問題点

PMI や t 検定といった共起頻度を用いる重み付け手法は「ある単語と共起しやすい単語は何か」というような、単語と単語の共起のしやすさの傾向を観察することによって文脈単語の重みを計算する。そのため、ある目標単語とある文脈単語がほかの単語と比較して共起しやすい傾向があれば、単語ベクトルの比較の際に重要な特徴になるとして、その目標単語に対する文脈単語により大きな重みを与えてしまう。図 3.1 は"scientist"という単語の PPMI と t 検定によって与えられた重みの上位 50 単語の例である。PPMI や t 検定において、"philosopher"や"researcher"といった、"scientist"とより関係していそうな単語により大きな重みを与えている一方、”asked”や”unable”といった、どの単語とも共起していてもおかしくない単語に対してもより大きな重みを与えてしまっている。特に”discovery”といったより”scientist”と関係していそうな単語が"recently"といった単語よりも小さな重みを与えてしまっている、共起頻度そのものの影響であるとしても、”recently”といったような基本的に単語ベクトルによって単語の意味を比較する際にあまり役立ちそうでない単語に大きな重み付けをしてしまうことは防ぐべきことである。

そこで、単語の意味の分別において重要な文脈単語を選択するために必要な情報は共起情報だけではないと考えた。目標単語と共起する単語自体が、どれくらい具体的な意味を持っているか、あるいはトピックに依存した単語なのかも共起性同様重要である。たとえ、共起に必然性があっても、単語ベクトルの類似

性を比較する際に役立たない単語であるかもしれない。共起性を考慮するだけでは、文脈単語の適切な重み付けを行うことはできない。そう考え、次章では、共起性に基づいた重み付けと単語トピック特定性に基づいた重み付けを結合させた提案手法を紹介する。

engineer	4.890624
noted	4.020726
islam	3.935112
asked	3.861004
philosopher	3.84682
worked	3.686651
influential	3.571538
unable	3.524532
prominent	3.494379
question	3.484527
tried	3.474771
professor	3.465109
honor	3.382112
astronomer	3.351608
queen	3.347326
artist	3.316468
david	3.313709
joseph	3.305478
visited	3.305478
believe	3.245819
modified	3.234302
prize	3.222917
intellectual	3.21915
teacher	3.21166
swedish	3.175026
allows	3.139686
impact	3.125893
argued	3.119067
william	3.112287
famous	3.09222
researcher	3.079062
agricultural	3.056444
poet	3.053254
refused	3.021903
saint	3.009633
neutron	2.997512
numerous	2.994504
azerbaijani	2.979601
peter	2.967836
nine	2.956208
communist	2.93335
nuclear	2.911002
henry	2.889144
mathematics	2.862475
recently	2.852004
global	2.831385
1960s	2.796294
increasingly	2.776781
discovery	2.767165
recognized	2.767165
nobel	2.738861
japanese	2.725004
birth	2.711337
astronaut	2.702328
district	2.688965
earliest	2.67142
cover	2.628861
elected	2.620562
hundred	2.620562
9	2.620562

(a)PPMI

engineer	0.027593
noted	0.015306
philosopher	0.013982
worked	0.012857
islam	0.011954
asked	0.011501
artist	0.010562
influential	0.00988
prominent	0.009484
question	0.009434
famous	0.009352
astronomer	0.008788
believe	0.008301
study	0.008266
prize	0.008199
william	0.00772
agricultural	0.007488
numerous	0.007236
unable	0.006814
tried	0.006636
professor	0.006602
honor	0.006316
queen	0.0062
david	0.006089
joseph	0.006062
visited	0.006062
modified	0.005833
intellectual	0.005786
teacher	0.005762
swedish	0.005649
allows	0.005541
impact	0.0055
argued	0.005479
computer	0.005374
researcher	0.005361
poet	0.005285
refused	0.005195
saint	0.00516
neutron	0.005125
azerbaijani	0.005075
peter	0.005042
nine	0.005009
communist	0.004946
nuclear	0.004885
henry	0.004826
mathematics	0.004754
recently	0.004727
global	0.004672
1960s	0.004581
individual	0.004557
increasingly	0.00453
discovery	0.004506
recognized	0.004506
field	0.004455
nobel	0.004434
japanese	0.004399
birth	0.004365
astronaut	0.004342
district	0.004309
social	0.004296

(b)t 検定

図 3.1 目標単語が"scientist"の場合の与えられた単語の重みの上位 30 文脈単語

第４章 提案手法

前章までに、単語空間モデルの概要と既存の重み付け手法とその問題点を見てきた。前章において PMI や t 検定といった共起情報だけに基づいて行われる重み付け手法では、より具体的な意味を持つ単語が共起的な必然性を持つ抽象的な単語よりも軽視される場合があることについて説明した。そこで、目標単語と文脈単語の共起性だけを考慮するのではなく、その文脈単語そのものをつまり、その文脈単語自体がどれくらい具体的な意味を持っているかを考慮する重み付け手法を提案する。具体性という概念には様々な意味合いがあるが、本手法においては、特定のトピックに集中して出現していればいるほど、より具体的な意味合いを持った単語であるとする。これを単語のトピック特定性という、ある単語がどれくらいの割合のトピックにおいて使用されているかどうかを表現した性質によって定義する。単語の性質において、専門性の高い単語であれば、特定のトピックにおいてでしか出現せず、機能語や一般用語であれば、より多数のトピックにおいて使用される傾向がある。この性質をもとに、より広い範囲のトピックで用いられる単語の重みを小さく、より狭い範囲のトピックにおいてしか使用されない単語の重みを大きくする単語トピック特定性を定義した。この単語トピック特定性に基づいて重み付けを行えば、”also”や”another”といったより一般的な単語により小さな重みを、より”mining”や”informatics”といったより専門性の高い単語にはより大きな重みを与えることができる。

また、単語トピック特定性は、文脈単語においての重み付けをすることを目的にしている。目標単語が何であろうと文脈単語に対する重みは常に一定である。そのため、単語トピック特定性のみによって与えられる重みは単語ベクトルにとって何の意味を持たない。そこで、その単語同士の共起が有意であるかどうかを判断するために PMI や t 検定といった重み付け手法と単語トピック特定性に基づいた重み付け手法を組み合わせた重み付け手法を提案する。単語トピック特定性を Latent Dirichlet Allocation (LDA)[12]によって生成される単語に対す

るトピックの確率分布とトピックの確率分布と Jensen-Shanon ダイバージェンス[21]、つまり情報距離を用いることによって数学的に定義した。そして、得られた単語トピック特定性の値を PMI や t 検定のような共起性に基づいた重みの値と結合させた。

本章では提案手法の概要を示した後、それぞれのステップにおいて用いられる手法を理論背景とともに説明する。

4.1 提案手法の概要

提案手法の概要を図 4.1 に表わした。本手法において過程 1、過程 2 があり、それぞれの過程においての入力データおよび出力データは以下の通りである。なお、過程 1、過程 2 の前に訓練データを用いてトピックの最適数を決定する。

過程 1
入力データ:大規模コーパス
出力データ:単語トピック特定性(WTS)の評価値
過程 2
入力データ:大規模コーパス、単語トピック特定性の評価値
出力データ:文脈単語を各要素に持つ単語の意味ベクトルから形成されたベクトル行列

本手法の手順を説明する。まず Arun の手法[19]を用いて、最適なトピック数を決定した後、過程 1 を行う。過程 1 において、まず大規模コーパスに含まれる単語からストップワードを取り除いた文書集合を生成する。その文書集合から各要素に単語の文書頻度を登録した単語文書行列を生成する。その後、生成した単語文書行列に Latent Dirichlet Allocation (LDA)[12]を用いることによって、単語トピック行列と文書トピック行列を生成する。その後、単語トピック行列と文書トピック行列から求めたトピックの確率分布と分布の類似度を比較して、単語トピック特定性を計算するために Jensen Shannon ダイバージェンス(JSD)[21]を計算する。過程 2 において、まず過程 1 と同様に大規模コーパスに含まれる単語からストップワードを取り除き、順序を持った単語集合を生成する。その単語集合から各目標単語においての文脈ウィンドウ内で共起する単語と頻度を記録し、単語単語行列を生成する。そして、生成された単語単語行列の各要素において PMI や t 検定といった共起に基づく重み付け手法によって各行列の要素を重み付けする。その後、その重みを過程 1 によって出力した単語トピック特定性と組合せて、単語-単語行列の各要素に登録する。図 4.1 の過程によっ

て、単語の意味ベクトルが得られる。

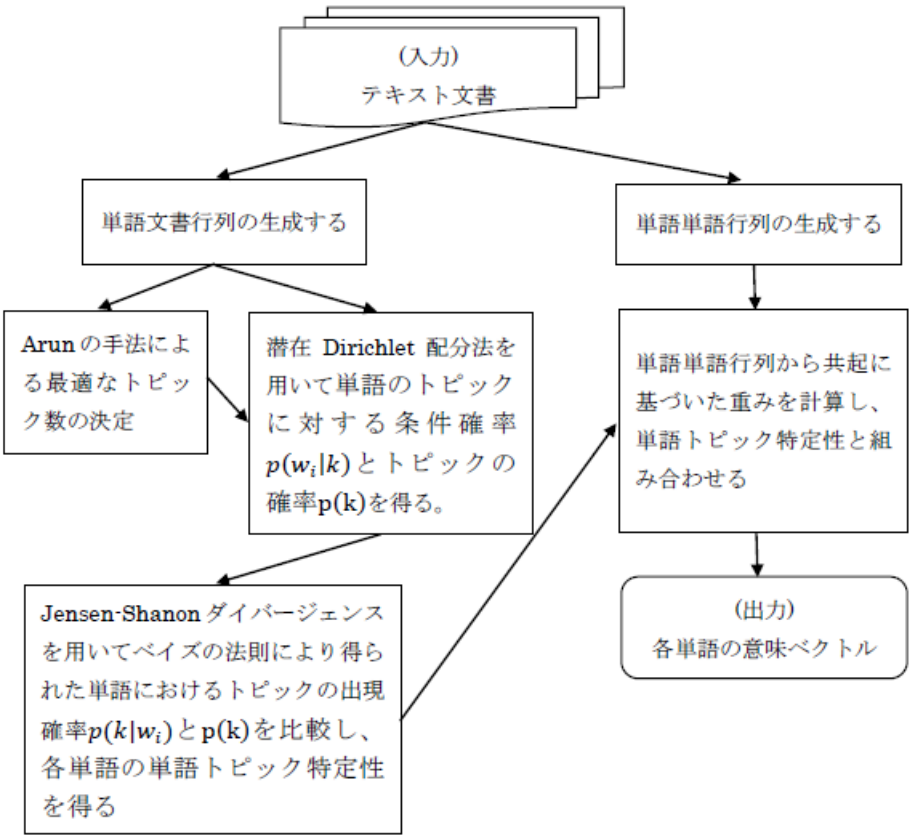

図 4.1 提案手法の概要

4.2 潜在 Dirichlet 配分法

潜在 Dirichlet 配分法などのトピックモデルは文書集合のような離散データにおいての潜在的なトピックを発見し、その発見されたトピックによって文書をモデル化する教師なしの統計アプローチである。ほとんどのトピックモデルにおいて文書を単語の順番を完全に無視する bag of words (BOW)形式で表現する[13]。つまり文書内で単語の交換ができるということを前提とする。この前提によって効率的にモデルにおいての計算をできる。また、こういったある意味粗末な前提によるモデルであるにもかかわらず、相対的により良いトピックを見つける傾向にある。我々は、このトピックモデルの中でも最も単純なモデルである LDA を用いることによって、コーパスから生成される単語文書行列(参照第 2 章)から、単語トピック特定性の計算に必要となるトピックの分布と単語に対するトピックの分布を計算する。また、LDA は実際のデータで実行する際、トピック数をあらかじめ決定しておかなければならない。LDA モデルではトピック数

は既知であることが前提となっているが、現実にそんな場合は存在しない。最適なトピック数を選定する必要がある。本研究においては Arun の手法[19]を用いることで最適なトピック数の選定を行った。

この節においてはまず、LDA の生成過程について説明し、その後 Gibbs サンプリングによる推論、最後に Arun の手法について説明する。

4.2.1 生成過程

潜在的ディリクレ配分法(Latent Dirichlet Allocation)は、最初の生成的トピックモデルであり、最も単純なトピックモデルの一つである[12][13]。

LDA において文書ごとにトピックの分布$\boldsymbol{\theta}_d = (\theta_{d1}, \dots, \theta_{dK})$があると仮定し、トピック分布$\boldsymbol{\theta}_d$によって文書 d におけるそれぞれの単語に対して、トピック z_{dn} が割り当てられる。その後、割り当てられたトピックの単語の分布$\boldsymbol{\Phi}_{z_{dn}}$によって単語が生成される。なお、トピックの単語分布$\boldsymbol{\Phi} = (\Phi_{k1}, \dots, \Phi_k)$はパラメータが$\beta$である Dirichlet 事前分布によって生成される。βが大きいほど、複数の単語が共起しやすくなる。同様に文書ごとにトピックの分布$\boldsymbol{\theta}_d = (\theta_{d1}, \dots, \theta_{dK})$はパラメータ$\alpha$ の Dirichlet 事前分布トピック分布から生成することができると LDA では仮定している。αが大きいほど、複数のトピックが共起しやすくなる。αやβは事前分布を制御するパラメータであり、ハイパーパラメータと呼ばれる。LDA モデルの生成過程は次のように記述される。w_{dn} は文書 d の n 番目の単語である。

K 次元のパラメータベクトルαの Dirichlet 分布を与える
V 次元のパラメータベクトルβの Dirichlet 分布を与える
for トピック 1 からトピック K
　　パラメータβによる Dirichlet 分布からトピック k に対しての多項分布Φ_kを決定する
for 文書 1 から文書 D
　　パラメータαによる Dirichlet 分布から文書 d に対しての多項分布θ_dを決定する
　　for 文書中の単語(単語 1 から単語 N_d)
　　　　θ_dからトピック $z_{d,n}$ を決定
　　　　Φ_{zn}から単語 $w_{d,n}$ を決定

Dirichlet 分布確率密度関数$p(\theta|\alpha)$は$\alpha = (\alpha_1, \alpha_2, \dots, \alpha_K)(\alpha_k > 0)$をパラメータとして以下のように定義される。

$$\mathrm{Dir}(\boldsymbol{\theta}|\alpha) = \frac{\Gamma(K\alpha)}{\Gamma(\alpha)^K}\prod_{i=1}^{K}\theta_i^{\alpha-1} \quad (4.1)$$

ここで、Γは Gamma 関数を示している。Gamma 関数は階乗を一般化した関数であり、以下のように表現される。

$$\Gamma(\mathrm{z}) = \int_0^\infty t^{z-1}e^{-t}dt \ (4.2)$$

Dirichlet 分布は、指数族であり、有限次元十分統計量を持っていて、多項分布に対する共役事前分布である。これらの特性によって LDA に対する推論やパラメータ推定アルゴリズムを簡単にすることができる。

前述した生成過程によって以下のような条件付き確率が得られる。

$$\mathrm{p}(w, z, \theta, \Phi|\alpha, \beta) = \mathrm{p}(\Phi|\beta)\mathrm{p}(\theta|\alpha)\mathrm{p}(z|\theta)\mathrm{p}(w|\Phi_z) \ (4.3)$$

上式のそれぞれの因子について説明すると$\mathrm{p}(\Phi|\beta)$は単語分布Φがパラメータβを持つ Dirchlet 分布に依存することを意味し、$\mathrm{p}(\theta|\alpha)$は文書レベルのトピック分布$\theta$はパラメータ$\alpha$を持つ Dirichlet 分布に依存することを意味し、$\mathrm{p}(z|\theta)$は、トピック集合 z は文書レベルのトピック分布θから依存する、$\mathrm{p}(w|\Phi_z)$は単語集合 w は単語分布Φとトピック集合 z に依存することを意味する。前述した LDA の生成過程をグラフィカルモデルで表現すると図 4.2 のようになる。色がついている円は観測した変数を表現し、白円は未知の変数を表現する。矩形は取り囲まれたノードにおける繰り返しを意味する。繰り返しの回数は矩形の隣の小文字によって表記される[14]。N はドキュメント d におけるトークン数、D は文書数、K はトピック数である。

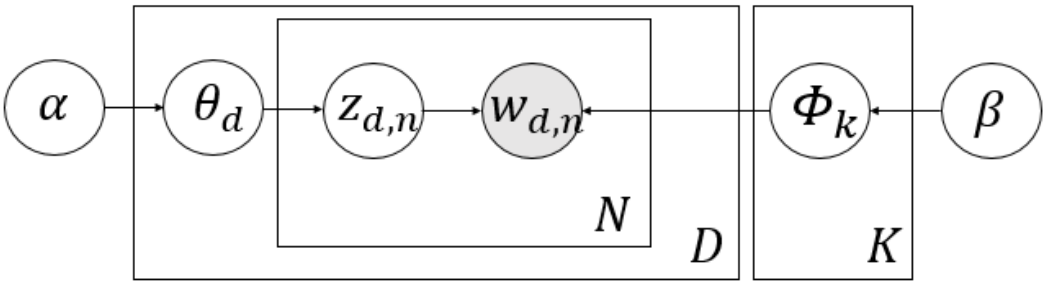

図 4.2 LDA のグラフィカルモデル

4.2.2 ギブスサンプリング

トピックモデルにおいて事後推定は重要な問題である。事後推定とは定義された生成過程を逆転させ、観測されたデータを与えたモデルにおいての潜在変数の事後分布を学習することをいう。LDA においては、以下の式を解くことに等しい。

$$\mathrm{p}(\theta, \Phi, \boldsymbol{z}|\boldsymbol{w}, \alpha, \beta) = \frac{p(\theta, \Phi, \boldsymbol{z}, \boldsymbol{w}|\alpha, \beta)}{p(\boldsymbol{w}|\alpha, \beta)} \quad (4.4)$$

この分布は非常に複雑で正確な推論によって解くことができない。正規化係数である$p(\boldsymbol{w}|\alpha, \beta)$は正確に計算することができない。そこで、それらを近似的に推論する手法がいくつかある。LDA において、それらを推定する期待値最大化、変分近似法などの様々な手法があるが、本研究においては Gibbs サンプリングを用いる。Gibbs サンプリングは実装が簡単で、記憶容量をあまり必要としないからである[15]。

Gibbs サンプリングはすべての条件付確率$\mathrm{p}(x_i|x_{-i})$が既知である場合の条件付確率$\mathrm{p(x)}, \mathrm{x} \in \mathbb{R}^n$からのサンプリングに対する Markov 連鎖 Monte Carlo 法である。Gibbs サンプリングは、他の潜在変数と観測の状況によって条件づけられたそれぞれの潜在変数を繰り返しサンプリングすることによって事後分布を再現することができる。Gibbs サンプリングの更新式を定義すると以下のようになる[16]。

$$\mathrm{p}(z_i = j|z_{-i}, \mathrm{w}) = \frac{n_{-i,j}^{(w_i)} + \beta}{n_{-i,j}^{(\cdot)} + W\beta} \cdot \frac{n_{-i,j}^{(d_i)} + \alpha}{n_{-i,\cdot}^{(d_i)} + T\alpha} \quad (4.5)$$

ここで$n_{-i,j}^{(w_i)}$は現在の割り当てであるトピック i を含まないトピック j に割り当てられた単語w_iの事例数である。$n_{-i,j}^{(\cdot)}$は、現在の割り当てであるトピック i を含まない、トピック j に割り当てられた単語の数である。$n_{-i,j}^{(d_i)}$は現在の割り当てであるトピック i を含まない、トピック j に割り当てられた文書d_iの単語の数である。$n_{-i,\cdot}^{(d_i)}$は現在の割り当てであるトピック i を含まない文書d_iの合計の数である。α とβは経験分布の得られた分布の選択することができるハイパーパラメータである。式(4.5)の初めの比がトピック j におけるw_iの確率、2 番目の比が文書d_iにおけるトピック j の確率を表現している。事後分布p(z|w)から十分に反復を行うと、個々のトピックの内容とは無関係な統計量を計算することができる。Gibbs サンプリングの結果、パラメータ Φと θを推測することができる。

$$\hat{\Phi}_j^{(w)} = \frac{n_j^{(w)} + \beta}{n_j^{(\cdot)} + W\beta} \quad (4.6)$$

$$\hat{\theta}_j^{(d)} = \frac{n_j^{(d)} + \alpha}{n_\cdot^{(d)} + T\alpha} \quad (4.7)$$

$\hat{\Phi}_j^{(w)}$は、トピック j から引き出される単語 w に対する多項パラメータであり、$\hat{\theta}_j^{(d)}$は文書 d から引き出されるトピック j のパラメータである。

4.2.3　トピック数の選定

前節までで説明してきた LDA を実際のデータで動かす際にはトピック数を決めておかなければならない。そのトピック数によって LDA モデルにおけるトピックの分別性が影響される。例えば、トピック数が少なすぎてしまうと、トピックはとても抽象的になる。つまり、トピック同士が重複してしまい、トピック同士が類似したものとなってしまう。そのため、LDA によって生成されたトピックがあまり重要な意味を持たなくなってしまう。一方で、トピック数を最適なトピック数より多く設定してしまうと、トピックはより具体的になる。トピック数が多すぎるので、トピックの分布が単語に対してよりスパースになってしまい、単語とトピック間に強い相関が生じてしまう。このような状況になってしまうと、トピックに対する文書の事後分布推定を行うことができない。つまり、多すぎるトピック数を設定した LDA によって生成されたトピックは本来のデータを正確に反映することができない。これゆえ、最適なトピック数を設定することが非常に重要である。

最適なトピックの数を選定する方法はいくつか提案されている。一般的には、汎化能力を示す Perplexity を用いて、複数のトピック数設定で行った場合を比較し、最も Perplexity が小さかったトピック数を選択するという方法がある[14]。また、Dirichlet 分布を拡張させた Dirichlet 過程によって LDA において次元数の変更を可能にすることによって、自動的にトピック数を決定する手法も提案されている[20]。しかし、本研究においては、他手法よりもより強健な挙動をみせ、より実装が簡単である Arun によって提案された手法[19]を用いる。彼らは、ハイパラメータが一定の設定においてトピック数 K を変化させて LDA の処理を行い、それぞれの LDA の出力であるトピック-単語行列と文書-トピック行列から生成される分布を観察することによって、最適なトピック数を決定できると記述した。以下では Arun の手法によるトピック数の選択について説明する

ために、まず特異値分解について説明した後、手法の詳細について記述していきたい。

4.2.3.1　特異値分解（SVD）

特異値分解(SVD)は 3 つのより単純な行列の積に行列を分解する手法である。SVD は m×n の行列 A を以下のように因子分解する[17]。

$$A = U\Sigma V^T \quad (4.8)$$

V^Tは行列 V の転置である。U は m×m の直交行列であり、V は n×n の直交行列である。直交行列 U と V の行は正規直交であり、それぞれ、$U^TU = I$、$V^TV = I$である。行列Σは対角行列である。Σの対角要素は特異値と呼ばれる。$r = \mathrm{rank}(A)$が A の線形独立の列の最大数であると定義すると、行列Σは対角線上における初めの r 要素を除いてすべて 0 である。それらの値は非負実数値であり、$\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_r > 0$というように降順に並んでいて、行列$\Sigma$は以下のように表記できる。

$$\Sigma = diag_{m\times n}\{\sigma_1, \ldots, \sigma_r\}$$

上記において、$\sigma_1, \ldots, \sigma_r$は$AA^T$の固有値の平方根であまた、それらの要素を A の特異値と呼ぶ。特異値分解を図示すると以下のようになる。

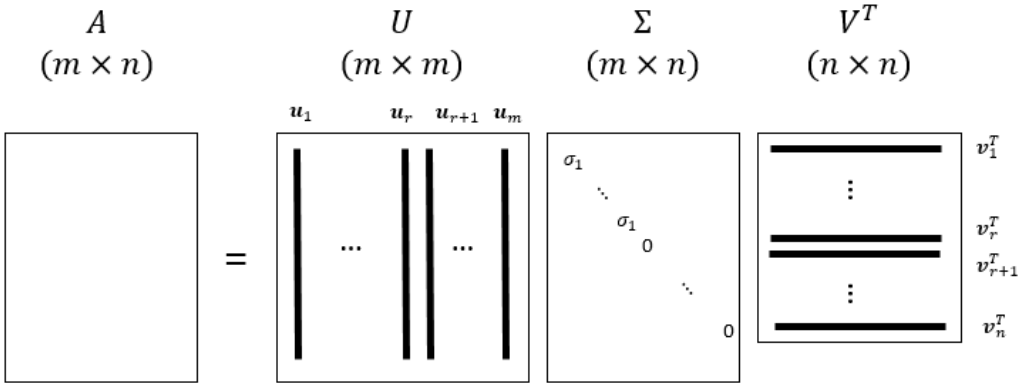

図　4.3 m×n の行列の特異値分解

また、$V^TV = I$であるから式(4.8)の両辺に Vを掛けることによって$AV = U\Sigma$が得られる。つまり、$Av_i = \sigma_i u_i$ (i = 1, … p)である。同様にして、$A^Tu_i = \sigma_i v_i$ (i = 1, … p)である。$Av_i = \sigma_i u_i$を幾何学的に解釈すれば、行列 A の特異値は、U の要素を主軸方向とした超楕円体$E = \{Ax: \|x\|_2 = 1\}$の半軸の長さである。つまり、特異値の分布は、長楕円体のそれぞれの方向における軸の分散とみることができる[18]。以下は特異値が 2 次元の楕円体の半軸となる例である。

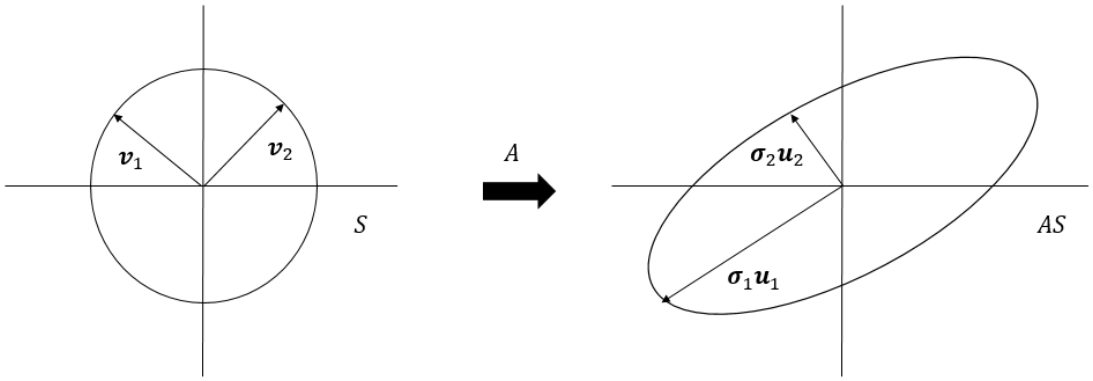

図 4.4 2×2 行列の特異値分解

4.2.3.2 Arun の手法によるトピック数の選択

前節で説明したように LDA モデルによってトピックに対する単語の確率と文書に対するトピックの確率を推定することができる。ここで、D、K、W をそれぞれ、文書数、トピック数、単語数とし、LDA を別の観点から見ると、K×W のトピック単語行列 M1 と D×T 文書トピック行列 M2 の非負値行列因子分解とみることができる。M1 における k 番目の列が k 番目のトピックにおける単語に対する分布を表現し、M2 における n 番目の列が n 番目の文書におけるトピックの分布を表現する。また、前節より、K×W 行列 M1 の特異値の分布はトピックにおける分散の分布とみなすことができる。

もしこの行列のトピックが単語に対してうまく分割されている、つまり、各トピックに割り当てられた単語をV_iとしたとき、$i \neq j$のとき、$V_i \cap V_j = \emptyset$ (i, j = 1, … , k)になるとすると、K×W 行列の特異値σ_iは K×W の i 列の L_2 ノルムと同等になる。これゆえ特異値に対しての分布が適切なトピックの数に達したとき、L_2 ノルムに対しての分布に十分近づくことを期待できる。ゆえにそれぞれのトピックが直交に近づく最初のトピック数を観察すれば、最適なトピック数を知ることができる。しかし、確率過密性によって特異値の分布と L_2 ノルムの分布を直接比較することができないため、行列 M2 から求められるコーパスのトピック分布を代用する。行列 M2 は文書におけるトピックの分布を表現しているので、単語おける分布を表現する M1 の特異値の分布と比較するのは正しくない。よって、行列 M2 とそれぞれの文書の長さを成分としたベクトル L の積をとり、特異値とそのベクトルの積の分布同士を比較する。

以上をまとめると、トピック数の選択は、すべての K に対して、以下の式のようにトピック単語行列と文書トピック行列の特異値間の対称 KL ダイバージェンス(4.3.1 にて後述)を計算することによって行うことができる。

$$\text{Measure}(\text{M1}, \text{M2}) = \text{KL}(C_{M1} || C_{M2}) + \text{KL}(C_{M2} || C_{M1}) \quad (4.9)$$

ここで、C_{M1}はトピック単語行列の特異値の分布である。C_{M2}は、L をコーパスにおけるそれぞれの文書の長さを成分とする一次元ベクトル、M2 を文書-トピック行列とした場合にベクトルの積 L×M2 を標準化することによって得られる分布である。W が十分に大きい場合、行列 M1 の特異値とコーパスにおいて存在する各トピックの割合を成分としたベクトルの分布は成分ごとに非常に良く似てくるので、式(4.9)は 0 に近づく。つまり、トピックの最適な数は、上記の尺度が最小値であるときのトピック数を選ぶことによって決定される。

4.3　単語トピック特定性

単語には特定の分野でしか用いられないものもあれば、幅広い分野や文書において用いられる単語もある。前者は単語ベクトルに意味弁別性を持たせるために役立つ文脈単語となるが、後者の単語は特定の分野に存在するのではなく、大抵の分野に存在するため、単語の意味を分別する際の有力な文脈情報となり得ず、ほとんど役に立たない。一般的にそういった単語は、データ処理を行う前に、人手によって取り除く。我々も機能語やストップワードと呼ばれる単語を前処理において取り除いてはいるが、それらの単語の除去を最小限にとどめ、一般的でないストップワードに関しては統計的に求められる重み付けによって重視するか重視しないかを決定する。前述した分野をLDAモデルにおけるトピック、つまり同じ文書で出現しやすい単語の集合のこととすると、LDA モデルにおいて、抽象的な単語は、どの単語においても共起しやすい結果、ほとんどすべてのトピックに出現する傾向がある。また、具体的な単語は特定の単語としか共起しないために特定のトピックにしか割り当てられない傾向がある。この LDA の性質に着目し、LDA によって割り当てられる単語のトピックの特定性から計算した文脈単語の有効性の指標を単語トピック特定性(Word Topic Specificity, WTS)と定義した。最も曖昧な単語はすべてのトピックに対して一様に分布すると仮定し、一様に分布すると仮定した際の単語のトピックに対する条件確率を次のように定義した。

$$p(k|w_{abstract}) = p(k) \quad (4.10)$$

上式の左辺は、後述するが LDA によって生成された$\hat{\Phi}_j^{(w)}$と$\hat{\theta}_j^{(d)}$によって得ることができる。なお、上式において、$\sum p(k|w_{abstract}) = 1$とする。

LDA によって得られる単語に対するトピックの条件確率$p(k|w_i)$と式(4.10)によって定義した最も曖昧な意味を持つトピックの分布の違いを数学的に求めるために、距離を用いる。一般的に距離といえば、Euclidean 距離や Mahalanobis 距離が有名であるが、そのような距離がすべての場合において最適であるとは

限らない。実際、Euclidean 距離はデータの分布と無関係であり、Mahalanobis 距離はデータの大域的な分布しか考慮出来ないためにそれら 2 つの距離は 2 つの確率分布間の距離尺度としては不適切である。また χ^2 検定[32]や尖度と歪度によって、分布の偏りを計算する方法[33]はあるが、その方法によって得られる値は、WTS が上位にくる単語を過大評価してしまうことが、予備の実験において分かった。そこで我々は 2 つの確率分布の距離を計算するために Jensen-Shannon ダイバージェンスを使用した。Jensen-Shanon ダイバージェンスは 2 つの異なる確率分布間の距離であり、非対称であり距離の公理を満たさない Kullback-Leibler ダイバージェンスを 2 つの確率分布の平均を取ったりすることによって対称にしたものである。この Jensen-Shanon ダイバージェンスを用いて、確率分布間の距離を計算すると、確率分布同士が類似しているほど、0 に近い値を取り、異なっているほど 1 に近い値を取る。つまり、式(4.10)による分布と比較することによって、単語トピック特定性のない単語は小さい値を単語トピック特定性のある単語は大きな値を与えることができる。

次節からは、Jensen-Shannon ダイバージェンスについて説明するために、まず、その構成要素である Kullback-Leibler ダイバージェンスについて説明した後、Jensen-Shannon ダイバージェンスに記述する。そして、単語トピック特定性と Jensen-Shannon ダイバージェンスの関係性について説明する。

4.3.1 Kullback-Leibler ダイバージェンス(KLD)

相対的なエントロピーとは 2 つの確率分布間の距離の尺度である。様々なダイバージェンスが分布間の類似度の尺度として定義されてきたが、最も重要なダイバージェンスの一つとして Kullback-Leibler ダイバージェンス(KLD)がある。KLD は 1951 年に Kullback と Leiber によって提案された 2 つの確率分布がどれくらい違っているかを表現する一般的な距離関数である[21]。統計分野における尤度比の期待対数として示される。この関数は古典的統計理論においては交差エントロピーや有向ダイバージェンスとして知られ、相対的な不確実性を測る。KL ダイバージェンスは Q から P の理論的な距離の非対称の情報尺度である。有限集合 χ における P と Q の分布の KL ダイバージェンスは以下のように定義される。

$$\mathrm{KLD(P||Q)} = \sum_{x \in \chi} P(x) log \frac{P(x)}{Q(x)} \quad (4.11)$$

上記の関数が相対的に小さいと逆に 2 つの変数の分布がより類似していること

になる。KLD はほとんどの場合において非負であり、分布 P と分布 Q が同じ、つまり P=Q のときゼロである。KLD は対称ではなく、距離の公理を満たしていないため分布間の真の距離ではない。このため、統計距離や擬距離と呼ばれることもある。また、KLD は Q=0 で P≠0 の場合の値が定義できない。つまり、KLD(P||Q)を定義するためには、P>0 のとき必ず Q>0 でなくてはならない。KLD は非対称であるが以下のように KL ダイバージェンスを対称にした例もある[22]。

$$\mathrm{SKLD(P||Q)} = KLD(P||Q) + KLD(Q||P) \quad (4.12)$$

4.3.2 Jensen-Shannon ダイバージェンス(JSD)

Jensen-Shannon ダイバージェンス(JSD)は対称で、有限である。2 つの確率分布間の JSD は平均分布に対するそれぞれの分布の KLD の平均として定義される。JSD は以下のように定義される[23]。

$$\mathrm{JSD(P||Q)} = \frac{1}{2}KLD(P||\frac{1}{2}P + \frac{1}{2}Q) + \frac{1}{2}KLD(Q||\frac{1}{2}P + \frac{1}{2}Q) \quad (4.13)$$

2 つの分布間の JSD の値は類似性がなくならばなくなるほど増加し、異なる要素に対して分布の確率の大きさが集中しているときに最大となる。また、KLD は非負であるので、JSD も非負である。JSD の最大値はどの基底の対数を用いるかによって変わる。自然対数、つまり、基底が e の対数を用いると、0≤JSD(P||Q)≤log2 になる。JSD は基底 2 の対数を使うと 0≤JSD(P||Q)≤1 となる。

4.3.3 単語トピック特定性

LDA において、単語のトピックに対する条件確率$p(w_i|k) = \hat{\Phi}_k^{(w_i)}$、$p(k|d) = \hat{\theta}_k^{(d)}$が得られる。また、LDA モデルにおけるトピックの確率は

$$p(k) = \sum_d p(k|d)p(d)$$

であるから、以下のように定義できる。

$$p(k) = \frac{\sum_{d=1}^{D} \hat{\theta}_k^{(d)} N_d}{N} \quad (4.14)$$

ここで、N はコーパスにおけるすべてのトークン数であり、N_d は文書におけるトークン数である。ベイズの法則を用いると、単語のトピックに対する条件確率$p(w_i|k)$とトピックの確率p(k)からトピックの単語に対する確率を計算すること

$$p(k|w_i) \propto p(w_i|k)p(k)$$

つまり、単語のトピックに対する条件確率$p(w_i|k)$を以下のように表現することができる。

$$p(k|w_i) = \frac{p(w_i|k)p(k)}{p(w_i)} \quad (4.15)$$

単語トピック特定性において最も機能的あるいは意味を持たない単語は、単語におけるトピックの出現確率分布とトピックの確率分布が同一であると仮定した。

単語トピック特定性を求めるために、単語におけるトピックの出現確率を P、典型的な機能的な意味を持つ単語のトピックに対する条件確率を Q とすると、それぞれ以下のような式になる。

$$P = p(k|w_i) \quad (4.16)$$

$$Q = p(k) \quad (4.17)$$

式(4.16)、式(4.17)で示される確率分布同士を比較するために前節で説明した Jensen-Shannon ダイバージェンスを用いる。JSD の値は、分布 P と分布 Q が近ければ、つまり、より典型的な機能的な意味を持つ単語の分布と近いほど、より小さな値をとり、遠いほどより大きな値をとる。例えば、単語トピック特定性の定義において、抽象的な単語と具体的な意味を持つ単語の分布はそれぞれ図 4.5、図 4.6 のようになる。図を見ればわかるように、抽象的な単語はより多くのトピックに満遍なく割り当てられており、具体的な単語は 1、2 のトピックにおいてだけ分布しており他のトピックの確率は 0 に近い。抽象的な単語は各トピックに一様に分布するという典型的な機能語の分布に非常に近いが、具体的な単語は非常に遠い。図 4.5、図 4.6 の例において、それぞれの図の分布を持つ単語の単語トピック特定性は、抽象的な意味しか持たない単語においては、0. 0009 であるが、具体的な意味を持つ単語は 0. 4538 である。このように単語トピック特定性は、より曖昧な意味を持つ単語に対しては 0 に近い値を、より具体的な意味を持つ単語に対してはより 1 に近い値を割り当てる。

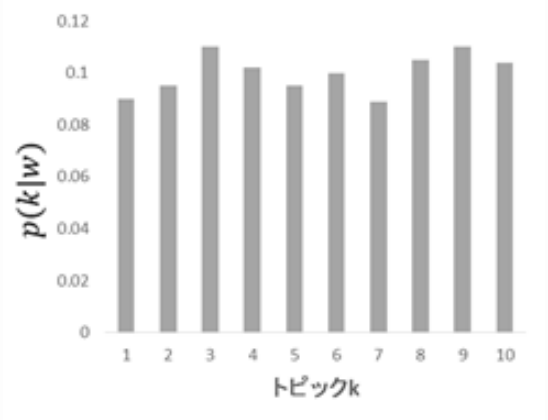

図 4.5 抽象的な単語のトピックに対する確率分布

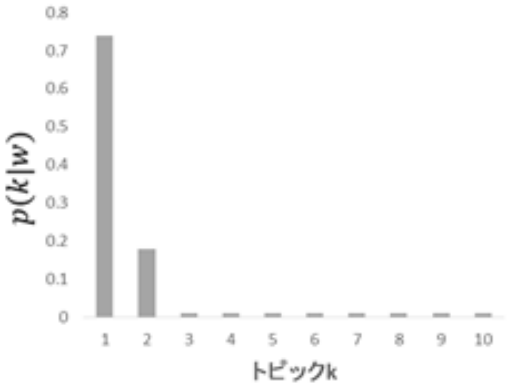

図 4.6 具体的な単語のトピックに対する確率分布

4.4 共起依存の重みとトピック依存の重みの融合

前節では、単語トピック特定性について説明した。その指標を用いれば、特定のトピックでしか用いられないもの単語により大きな値を与えることができ、幅広いトピックにおいて用いられる単語にはより小さな値を与えることができることを説明した。この重み付け手法を使えば、より具体的な意味を持つ文脈単語要素を重視することができる。しかしながら、本研究においては単語のトピック特定性だけに重点を置くのではなく、単語同士の共起性をも重視した重み付けを行いたい。特定のトピックにしか現れない単語だけが、単語の意味ベクトルの弁別性を高めるのに必要となるとは限らないからである。トピックに対しては抽象的であっても、単語の機能の観点からみると、ある単語がどの品詞であるかを特定する際に役立つかもしれない。実際、Turney は、単語合成を行う際、単語のトピック性と機能性を分けた行列を提案し、将来的にはトピック性、機能を示す単語、質を表わす単語、マナーを示す単語に分けた行列を提案することに

より、より単語の意味を柔軟に捉えようとした[23]。また、単語トピック特定性は共起単語の性質そのものを表現するものであり、目標単語の性質とは全く関係がないからである。単語トピック特定性による重みは文脈単語だけに関する重みであるため、目標単語と文脈単語の関係性を考慮することができない。つまり、単語のトピック特定性だけによる重みだけでは、関連した単語を見つける要素の重み付けとして不十分なのである。これゆえ、トピック特定性だけによる重み付けを行うのではなく、共起性にも基づいて重み付けを行いたい。そこで、以下では、共起性に基づいた重み付けと単語トピック特定性に基づいた重み付けを組み合わせる方法について説明していきたい。

4.4.1 単語トピック特定性の調整

単語トピック特定性の値をそのまま文脈単語の重み付けに対して用いると、様々な不都合が生じる。WTS は、LDA によって生成されるトピックに依存する。しかし、LDA は、より頻度が大きい単語をより多くのトピックに割り当ててしまうという問題がある。抽象度の高い単語は共起する単語が多くなるため、より多くのトピックに割り当てられると前述したが、高頻度語も同様に共起語が多くなるため、より多くのトピックに割り当てられやすいという傾向がある。つまり、WTS は、単語 A と単語 B がたとえ同じ具体性を持っていようと、単語 A がより多く出現するコーパスにおいて、単語 A の方により小さい値を、単語 B の方により大きな値を与えやすい。例えば、”automobile”と”car”という同義語がある。あるコーパスにおいて”car”という単語が”automobile”に比べて非常に多くの文書に登場すると、LDA の性質により、”car”の方がより多くのトピックに割り当てられる。そのため 2 つの単語はほぼ同等の具体性を持っているにも関わらず、WTS の差が頻度の影響によって大きくなってしまう。また、一般的に固有名詞は、低頻度語であるため、WTS の上位に固有名詞が集まってしまう。そのため、たまたまそういった単語と共起した単語のベクトルが、固有名詞などの特殊な文脈単語に対する重みを WTS によって不当に引き上げられてしまうために性質が劣化してしまう。実際、同義語の単語トピック特定性を比べると次のように頻度が大きいほど、単語トピック特定性の値は小さくなり、頻度が小さいほど、単語トピック特定性の値は大きくなる。

そこで WTS のコーパスにおける各単語の文書頻度の影響を最小限にとどめるために、より頻度が高い単語の WTS を引き上げて、より頻度の小さい単語の WTS を引き上げるために次のような式を定義した。

$$AdjustedWTS(c_i) = \big(WTS(c_i)\big)^{\alpha}\left(\frac{|d_i|}{|D|}\right)^{(1-\alpha)} \qquad (4.18)$$

$WTS(c_i)$は文脈単語c_iに対する単語トピック特定性、$|d_i|$は文脈単語c_iの文書頻度、$|D|$はコーパスにおける総文書数である。αは定数であり、最善のベクトルを生成するときの値であるとする。αが大きいほど WTS を重視する。また、$|d_i|/|D| \leq 1$であるから、$1-\alpha$が大きいほど、$|d_i|/|D|$は小さくなり、αを変化させることによって、文書頻度の影響の強さを調節する。

4.4.2 重み付けの結合

共起に基づく重みを PMI、t 検定をそれぞれの単語に対して計算した後、2 つのアプローチによって共起に基づく重みと単語トピック特定性に基づく重みを結合させた。これらのアプローチによる結合によって、共起性を考慮するだけでなくより具体性の高い単語はより重視する重み付けを行うことができるようになる。

4.4.2.1 積による結合アプローチ

積による結合アプローチでは、最も具体的な単語の共起に基づく重みは変化させず、単語の抽象度が増すほど、共起に基づく重みをより小さく評価したい。つまり、$WTS(w_C) = 1$であれば、共起に基づく重みを変化させず、WTS が小さくなるほど共起に基づく重みをより小さく評価したい。現実のデータセットにおいては$0 < WTS(i) < 1$ $(\mathrm{i} = 0,1,\dots,\mathrm{W}, \mathrm{W}$ は文脈語の数$)$であるから、積を取れば、具体的な単語をより評価できることとなる。積による結合アプローチでは以下の式によって共起に基づく重みと単語トピック特定性に基づく重みを結合される。

$$weight(w_T, w_C) = WC(w_T, w_C) \times AdjustedWTS(w_C) \quad (4.19)$$

上式において WC は共起性に基づく重みであり、WTS は単語トピック特定性に基づいた重みを式(4.18)で調整したものである。w_Tと,w_Cはそれぞれ目標単語、文脈単語を示す。

4.4.2.2 和による結合アプローチ

和による結合アプローチでは、和のアプローチと同様に$WTS(w_C) = 1$であれば、共起に基づく重みを変化させず、WTS が小さくなるほど共起に基づく重みをより小さく評価したい。現実のデータセットにおいては、$0 < WTS(i) < 1$ $(\mathrm{i} = 0,1,\dots,\mathrm{W}, \mathrm{W}$ は文脈語の数$)$であるから、WTS の対数を取ると、負になるので、その値を共起に基づく重みに加えると、具体的な単語をより評価できる。和による結合アプローチでは、以下の式によって共起に基づく重みと単語トピック特

定性に基づく重みを結合させた。

$$weight(w_T, w_C) = WC(w_T, w_C) + \log(AdjustedWTS(w_C)) \quad (4.20)$$

$$\text{positiveweight(x, y)} = \begin{cases} weight(w_T, w_C) \; if \; weight(w_T, w_C) > 0 \\ 0 \quad otherwise \end{cases} \quad (4.21)$$

式(4.20)において WC は共起性に基づく重みであり、WTS は単語トピック特定性に基づいた重みを式(4.18)で調整したものである。また、式(4.21)は、weight が 0 以上の場合のときだけの重みを表わす positiveweight である。式(4.20)において、WC が PMI の場合、次の式が成り立つ。

$$\begin{aligned} weight(w_T, w_C) &= PMI(w_T, w_C) + \log\big(AdjustedWTS(w_C)\big) \\ &= \log \frac{p(w_T, w_C)}{p(w_T)p(w_C)} + \log\big(AdjustedWTS(w_C)\big) \\ &= \log\left(\frac{p(w_T, w_C)}{p(w_T)p(w_C)} \times AdjustedWTS(w_C)\right) \end{aligned}$$

つまり、PMI の対数の中身と WTS の積を取ったものとなる。

第５章 評価

本章では単語トピック特定性を考慮した重み付け手法を評価するための実験結果について記述する。本研究の目的は、トピックモデルを用いて単語のトピック特定性による重みを計算し、その重みを共起に基づいた重みと結合することによって、共起に基づいた重みのみの単語ベクトルよりもより洗練された単語ベクトルを生成することである。その目的を押さえながら、以下ではまず、データセット、評価データセットについて述べ、その次にそのデータと単語ベクトル同士の比較によって得られる類似度のデータとの比較を行うための手法、最後にその比較手法によって得ることができるそれぞれの手法の評価に関する結果について述べる。

5.1　データセット

5.1.1 Wikipedia

Wikipedia はオープンアクセスできる、様々な分野の記事を含むインターネット百科事典である。Wikipedia は記事数が多く、記事がカテゴリ分けされているため、多くの自然言語処理の実験で用いられる。

本実験を行うあたり 2010 年の 4 月時点における Wikipedia のすべての英文記事のスナップショットから作成された The Wesbury Lab Wikipedia corpus というデータセット(コーパス)を用いる[24]。コーパスはタグなしで生のテキストである。また、2000 文字未満の文書は削除されている。以下がコーパスの詳細である。

コーパスサイズ:　約 10 億語、200 万文書以上
データサイズ:　6Gb 以上

上記のデータセットから無作為に 10 万文書を抽出し、訓練データセットとした。

なお、コーパスにおいて含まれる特殊文字や記号さらに単語ベクトルを生成する際の障害となるストップワードは前処理において除去した。

5.2　評価セット

提案手法の単語ベクトルの弁別性を評価するために人手によってスコア付けされた単語類似度の評価データセット、WordSimilarity-353、MEN-3000、MTURK-287、Rare Word Similarity (RW-2034)、RG-35 を用いた。以下にそれらの特徴についての詳細を記述する。

5.2.1 WordSimilarity-353(WS-353)

WordSimilarity-353[25]は、13 人の注釈者によって注釈付けされた類似度の平均と英語の単語のペアの 2 つのセットを含んでいる。全部で 353 単語ペアから構成される。注釈者が単語ペアの評価付けを行う際、単語ペア同士にどれくらい意味的に関連性があるかを絶対的に評価する。単語ペアにおける関連性を 0 から 10 までのスケールによって表現する。0 が全く関係していない単語ペアを意味し、10 が非常に関連性のあるまたは、全く同一の単語を意味している。そして最終的な評価値は各注釈者によって評価された評価値の平均によって表現される。

5.2.2 MEN-3000

MEN-3000[26]は、Amazon の Mechanical Turk を使うクラウドソーシングによって注釈付けされた類似度と英語の単語のペアの 2 つの集合を含んで利いる。テキストベースの意味的スコアに応じて関連性のレベルのバランスの取れた範囲を確保するためにサンプリングされたペアである ESP タグとして出現する無作為に選択された 3000 単語のペアから構成されている。注釈者が単語ペアの評価付けを行う際、2 つの単語ペアのどちらの方がより類似したペアなのかを比較することによって単語のペアの類似度を評価する。それぞれの単語ペアに対して、50 単語ペアを比較し、最終的には 50 ポイントスケールの評価値が得られる。評価値を 50 で割ることによって評価値は 0 から 1 に標準化される。

5.2.3 MTURK-287

割り当てあたり 50 単語ペアのバッチを Amazon の Mechanical Turk のクラウドソーシングによって評価することにより得られた 287 単語のペアから構成される[27]。最大 1 バッチ当たり 30 人の従事者によって行われ、各単語ペアを 23 人の Mturk の従事者の平均にすることで評価する。また、低品質な評価を考慮しないように校正基準を用いている。単語ペアにおける関連性を 0 から 10 までのスケールによって評価する。

5.2.4 Rare Word Similarity (RW-2034)

全部で 2034 単語ペアから構成されている。WS-353 や MEN-3000 のようなほとんどの単語ペアの類似度に対する評価データセットは、どの文書データに対しても一般的な単語しか含まれていない。単語ベクトル同士の類似度の比較のタスクにおいて、頻出する単語に対しては、精度はよいが、特殊な単語に対しては精度が悪い。一般的に、単語同士の類似度で重視されるべきものは特殊な単語である。明示的でない類義単語を抽出するためである。この評価データセットは主に特殊な単語から構成されている[28]。単語ペアを作る際 Wikipedia 内において珍しい単語同士のペアを自動的に作成し、その単語ペアに対して人間が評価付けを行った。こうすることによって、特殊な単語に対する評価をできるようになった。また、単語ペアにおける関連性を 0 から 10 までのスケールによって評価する。

5.2.5 RG-65

RG-65 は同義語または関連性のない単語、65 名詞のペアを含む評価データセットである[29]。51 人の注釈者によって注釈づけされ、0 から 4 のスケールで評価されている。

次に各データセットの性質をまとめた表を表わす。

表 5.1 評価データセット

	単語対	評価値
WS-353	353	0-10
MEN-3000	3000	0-1
MTURK-287	287	0-10
RW-2034	2034	0-10
RG-65	65	0-4

5.3 評価手法

本実験では、それぞれの重み付け手法が単語ペアの類似度を得るためにどれくらい良い表現を得ているのかどうかでそれぞれの手法に関する優劣を判断したい。例えば、「紅茶」と「コーヒー」は空間ベクトル上において近くに存在し、「紅茶」と「エンジン」は遠くに存在するような単語空間ベクトルにより大きな評価を与えたい。こういう評価を行うために前節で記述した人間が評価付けを行った単語ペアのデータセットとそれぞれの手法によって得られたベクトルの単語ペア間の類似度の関係性を評価する。前節で説明したように例えば、WS-353 や MEN というデータは、単語ペアの類似度が高ければより 10 に近い値が与えられており、全く関連性がないと 0 に近い評価を与えられている。このデ

ータセットと、各ベクトル同士を比較することによって得られる類似度を比較することによって、前述したような評価を与えることができる。

その評価を行う際、最も一般的であるのが各データセットの本来の値同士の関連を見るのではなく、それらの値を各データセットにおける順序変数にして 2 つのデータセットにおける変数の順序の関連性を見ることである。このようなノンパラメトリックな評価テストにおいて、最も一般的な手法が Spearman の順位相関係数である[30][31]。Spearman の順位相関係数による手法は、ノンパラメトリックな手法であるために母集団の分布に影響されない。つまり、データの順位を使うため、外れ値に対して鈍感であり、データが規則正しい空間によって収集される必要がない。よってとても小さな標本サイズであっても簡単に適用することができる。単語ベクトルは、ベクトルを形成する際のコーパスに非常に依存しており、規律正しい空間とは言えない。これゆえ、多くの単語ベクトルの生成の評価テストにおいて、Spearman の順位相関係数が用いられる。これゆえ、我々もこの順位相関係数を使用することとする。

Spearman の順位相関係数 r は評価値の順位間の相関を表現する。まず、2 つの変数 X と Y があるとすると、それぞれの変数における順位を最高から最低までの値の順序に並べることによって決定する。その後、それぞれの事例 i に対して、変数 X の順位と変数 Y の順位の違いを計算し、d_iとして定義し、2 乗誤差を$\sum d_i^2$と定義する。すると変数 X と Y 間の Spearman の順序相関係数は以下の式によって計算することができる。

$$r_s = 1 - \frac{6\sum d_i^2}{n(n^2-1)} \quad (5.1)$$

n はデータの数を表わす。r_sの値は-1 から 1 までである。1 は 2 つのデータセットにおける評価値の順位間の完ぺきな正の相関を意味する。r_sの値が 1 に近づくほど、2 つのデータセットの順番間の相関が大きくなるということであり、r_sの値が 0 に近づくほど、2 つのデータセットにおける順位の相関もないこと表わしている。r_sの値が- 1 に近づくほど、データセットの順位間に負の相関があることを示す。また、もし 2 つ以上同じ値が見つかったら、順位の平均を取る。例えば、1、2、3、4.5、4.5、6 などのようにする。

Spearman の順位相関の例について述べる。以下の表のような変数 X と Y があったとする。

表 5.2 変数 X と Y に対する順位と 2 乗誤差

事例	変数		順位		d_i	d_i^2
	X	Y	X	Y		
1	0.78	0.69	2	1	1	1
2	0.45	0.35	5	5	0	0
3	0.80	0.68	1	2	−1	1
4	0.25	0.14	7	7	0	0
5	0.35	0.48	6	4	2	4
6	0.11	0.08	8	8	0	0
7	0.46	0.28	4	6	−2	4
8	0.68	0.57	3	3	0	0
$\Sigma\ di^2$						10

上記の表より、Spearman の順序相関係数を計算すると以下のようになる。

$$r_s = 1 - \frac{6 \cdot 10}{8(8^2 - 1)} = 0.880952$$

このように Spearman の順位相関係数を用いることによって 2 つの評価値集合の順位間の相関を表現することができる。これによって、単語ベクトル同士を比較することによって得られるコサイン類似度の集合と評価セットにおける単語対の類似度集合の順序の相関を表現することができる。本実験では Spearman の順位相関係数によって単語ベクトル同士を比較することによって得られるコサイン類似度の集合と評価セットにおける単語対の類似度スコアの集合を比較し、相関係数が高いほど、人間の評価に近い類似した単語を取り出せるベクトル空間を得られたとしてより高い評価を与えることとする。

5.4 実験結果

5.4.1 実験 1: Arun の方法によるトピックの最適数の決定

まず、潜在 Dirichlet 配分法において設定するトピック数を決定するために Arun の手法を用いた。ハイパーパラメータ α=0.1 とし、β=0.1 とした。ギブスサンプリングの反復数は 2000 とした。図 5.1 を見ると分かるように、トピック数が比較的小さい場合、KL ダイバージェンスが小さい。そして、KL ダイバージェンスは、トピック数が 12 のとき最小となる。したがって、最適なトピック数は 12 となった。

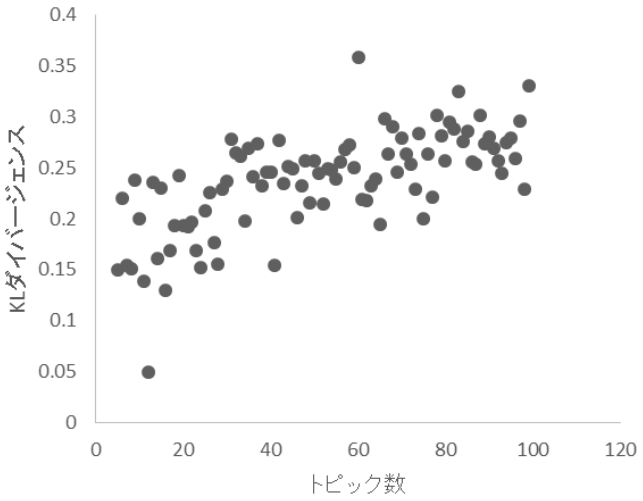

図 5.1 Arun の方法によるトピック数の決定

また、トピック数が最適な場合、つまり 12 の場合とトピック数が 30 の場合の PPMI と WTS の結合(積の結合アプローチ)の Spearman の順位相関係数を比較すると表 5.2 のようになる。なお、ウィンドウサイズは 4 とした。

表 5.3 トピック数による Spearman の順位相関係数の違い

topic	WS	MEN	MTURK
30	0.536	0.690	0.706
12	0.540	0.696	0.709

つまり、最適のトピック数に設定したときの LDA により求めた WTS による重みを考慮したベクトルの方が弁別性に優れている。

5.4.2 実験 2: 積の結合アプローチによる重み付け

本実験では前述した The Wesbury Lab Wikipedia corpus から得られた 10 万文書のデータから単語ベクトルを学習する。各単語ベクトルを学習する際、ウィンドウサイズは 1 から 10 に変化させた。我々は、2 つの単語のベクトル間のコサイン類似度を計算することによって、単語間の類似度を求めた。前節で説明した注釈者による評価値に対する提案手法の重み付けによって生成される単語間の類似度の評価値間の Spearman の順序相関係数を計算することによって提案手法によって得られたベクトルの弁別性を評価した。WTS を計算する際の定数は、1 万文書で実験した際に、Spearman の順序相関係数の最も評価値が高かったものとした。共起に基づく重みが PPMI の場合、$\alpha = 0.91$とした。表 5.3 に各実験結果をまとめた。また、単語トピック特定性による重みを求める際のトピックを Gibbs サンプリングに関してはハイパーパラメータα=0.1 とし、β=0.1 とし

た。反復数は 2000 とした。図 5.2、図 5.3、図 5.4、図 5.5 では評価データセット MS、MEN においての各実験の結果をグラフによって表現した。なお Freq は重み付けなし、つまり頻度だけの単語ベクトルのことである。各評価データセットに存在する単語対のうち、WS-353 では 256 単語対、MEN-3000 では 1349 単語対、MTURK-287 では 150 単語対、RW-2034 では 136 単語対、RG-65 では 20 単語対出現した。これら出現した単語対を評価のための単語対とした。また、RG の評価は基本的に不安定であるが、これは 20 単語対しかなく、標本数がとても少ないためである。よって、RG の評価データセットによる Spearman の順位相関係数の値は参考程度にする。

表 5.4 各手法における Spearman の順序相関係数(積のアプローチ)

window size	weighting	WS	MEN	MTURK	RW	RG
1	FREQ	0.326	0.360	0.387	0.284	0.335
	PPMI	**0.398**	0.550	**0.518**	**0.428**	**0.444**
	Ttest	0.445	0.560	**0.458**	0.463	0.340
	PPMI+WTS	0.381	**0.560**	0.491	0.402	0.432
	Ttest+WTS	**0.446**	**0.565**	0.448	**0.465**	**0.361**
2	FREQ	0.339	0.464	0.516	0.228	0.489
	PPMI	**0.488**	0.651	**0.660**	0.317	**0.550**
	Ttest	0.527	0.685	**0.627**	**0.414**	**0.451**
	PPMI+WTS	0.486	**0.666**	0.656	**0.325**	0.528
	Ttest+WTS	**0.531**	**0.687**	0.625	0.413	0.386
3	FREQ	0.362	0.490	0.574	0.217	0.525
	PPMI	0.524	0.673	**0.683**	**0.279**	0.514
	Ttest	0.580	0.723	**0.700**	**0.391**	0.496
	PPMI+WTS	**0.533**	**0.688**	0.682	0.245	**0.561**
	Ttest+WTS	**0.581**	**0.724**	0.698	0.383	**0.508**
4	FREQ	0.381	0.503	0.606	0.192	0.606
	PPMI	0.530	0.680	0.704	**0.242**	**0.624**
	Ttest	0.607	0.740	**0.745**	**0.375**	0.531
	PPMI+WTS	**0.540**	**0.696**	**0.709**	0.216	0.598
	Ttest+WTS	**0.609**	**0.741**	0.740	0.358	**0.535**
5	FREQ	0.381	0.505	0.605	0.180	0.620
	PPMI	0.518	0.679	0.700	**0.221**	**0.594**
	Ttest	**0.618**	**0.748**	**0.750**	**0.350**	0.504
	PPMI+WTS	**0.530**	**0.692**	**0.702**	0.202	0.568
	Ttest+WTS	0.615	0.748	0.744	0.336	0.504
6	FREQ	0.387	0.505	0.604	0.162	0.645
	PPMI	0.512	0.677	0.696	**0.193**	**0.624**
	Ttest	0.625	**0.751**	**0.745**	**0.347**	**0.546**
	PPMI+WTS	**0.532**	**0.690**	**0.707**	0.184	0.591
	Ttest+WTS	**0.627**	0.750	0.744	0.335	0.528
7	FREQ	0.383	0.508	0.605	0.156	0.654
	PPMI	0.505	0.677	0.696	**0.188**	**0.621**
	Ttest	0.625	**0.754**	**0.741**	**0.337**	**0.537**
	PPMI+WTS	**0.529**	**0.690**	**0.707**	0.157	0.595
	Ttest+WTS	**0.625**	0.753	0.739	0.322	0.525
8	FREQ	0.385	0.508	0.598	0.151	0.650
	PPMI	0.516	0.676	0.704	**0.183**	0.594
	Ttest	0.630	**0.756**	**0.741**	**0.346**	**0.552**
	PPMI+WTS	**0.538**	**0.690**	**0.711**	0.177	**0.633**
	Ttest+WTS	**0.630**	0.755	0.735	0.336	0.540
9	FREQ	0.386	0.511	0.599	0.154	0.642
	PPMI	0.521	0.675	0.706	**0.186**	0.585
	Ttest	0.634	**0.756**	**0.742**	**0.338**	**0.605**
	PPMI+WTS	**0.543**	**0.689**	**0.717**	0.177	**0.623**
	Ttest+WTS	**0.636**	0.755	0.739	0.329	0.549
10	FREQ	0.381	0.512	0.597	0.151	0.606
	PPMI	0.524	0.676	0.708	**0.172**	0.562
	Ttest	0.637	**0.757**	**0.741**	**0.317**	**0.580**
	PPMI+WTS	**0.546**	**0.688**	**0.714**	0.163	**0.595**
	Ttest+WTS	**0.640**	0.755	0.741	0.308	0.559

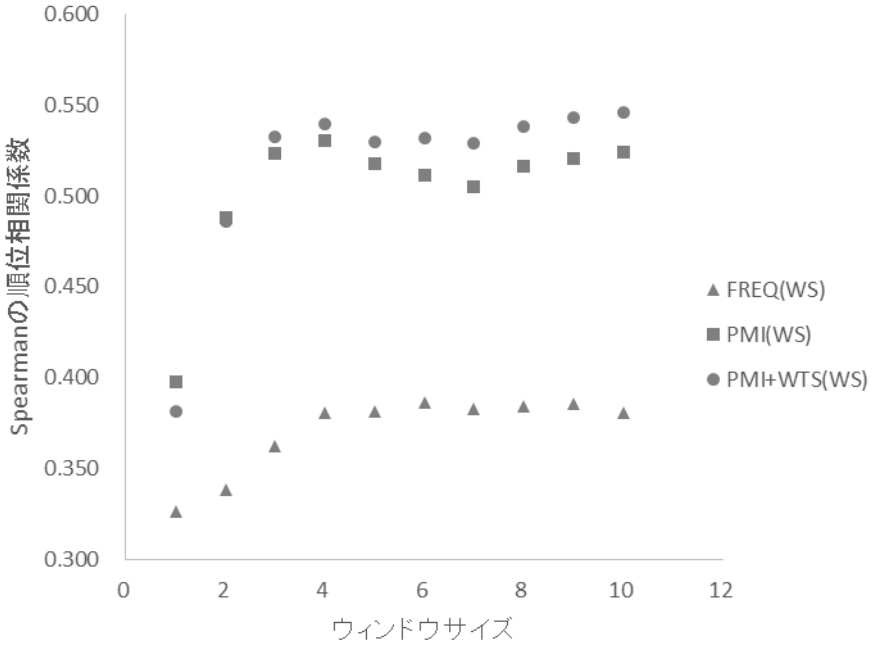

図　5.2 PPMI による重み付けと WTS を考慮した重み付けの比較(WS)

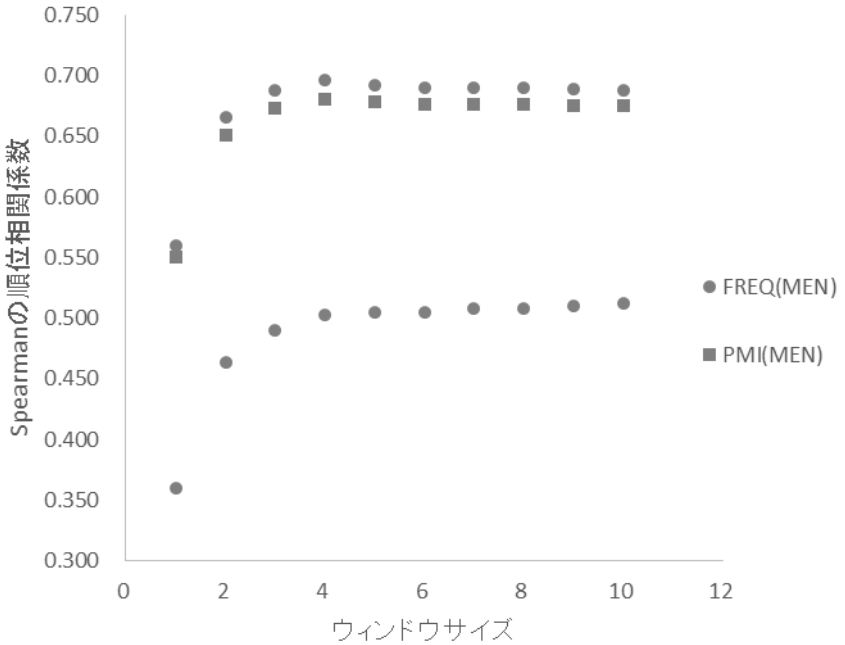

図　5.3 PPMI による重み付けと WTS を考慮した重み付けの比較(MEN)

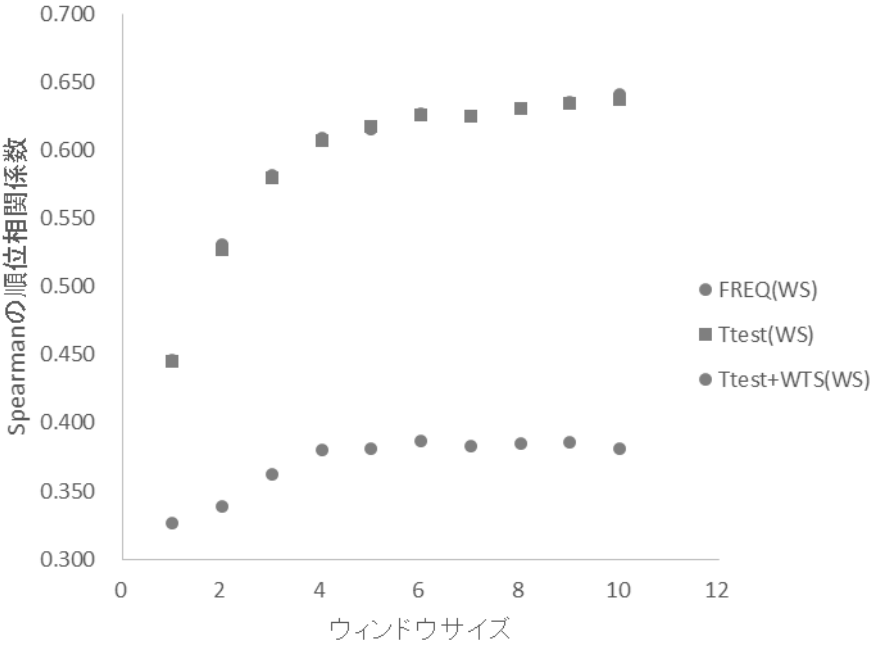

図　5.4 t 検定による重み付けと WTS を考慮した重み付けの比較(WS)

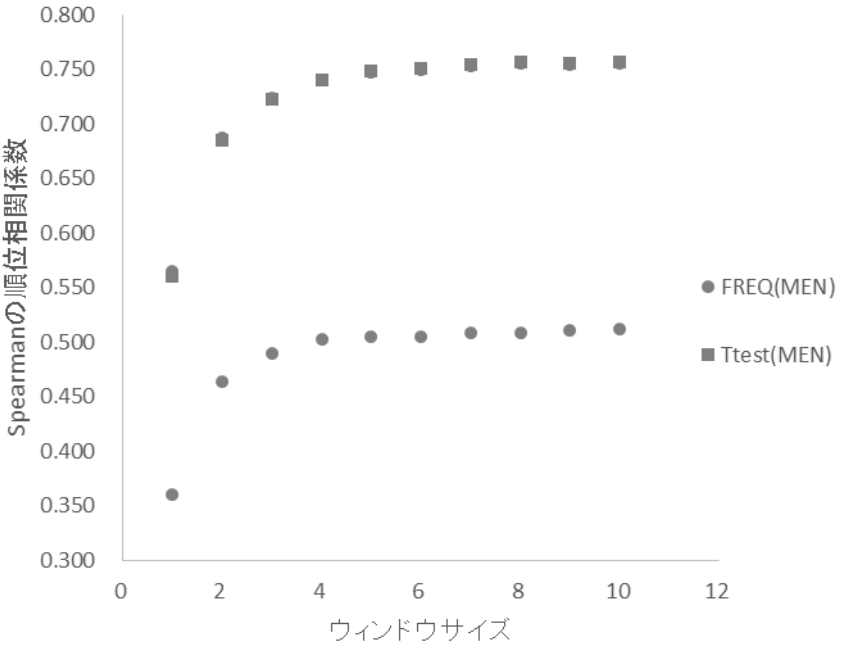

図　5.5 t 検定による重み付けと WTS を考慮した重み付けの比較(MEN)

ウィンドウサイズが小さい場合の Spearman 相関係数は、頻度だけの場合、共起のみ考慮した重み付けの場合、WTS も考慮した場合、いずれの場合においても小さかった。そしてほとんどの評価データセットの場合においてもウィンドウズのサイズが 4 から 5 のあたりのときから Spearman 相関係数の変化が乏しくなり、安定してきた。しかしながら、評価データセットが RW の場合、ウィンドウサイズが 1 のとき、Spearman 相関係数が最大になり、その後、ウィンドウサイズを大きくしていくほど、どの場合においても精度が悪くなっていった。

共起による重み付けが PPMI の場合、ウィンドウサイズが 1 で共起情報だけ考慮した重み付けを行ったときの Spearman 相関係数は、ほとんどの評価セットにおいて WTS も考慮したときと比較して大きくなっている。しかし、ウィンドウサイズを 1 として学習した際、前者の方が大きかったにもかかわらず、ウィンドウサイズ 2 によって学習した際は後者のほうが大きくなり、その後、ウィンドウサイズが大きくなるにつれて、後者と前者の Spearman 相関係数の差が広がっていった。そして、ウィンドウサイズが 6 あたりになるとその差が最大となり、WTS を考慮した場合の方が、共起性のみを考慮した場合と比較して 3%ほど相関係数が大きくなる。その後、ウィンドウサイズが大きくなるにつれて、その差が狭まっていった。よって、共起による重み付けが PPMI の場合、提案した重み付け手法がより弁別性があるベクトルを生成することが確かめられた。しかし、RW の評価セットの場合では、提案手法によるベクトルは既存の手法によるベクトルよりも弁別性が改善するどころか、劣化させてしまっている。

共起による重み付けが t 検定の場合、ウィンドウサイズに関わらず共起情報だけ考慮した場合と WTS も考慮した場合の Spearman 相関係数はほとんど変わらなかった。よって、共起による重み付けが t 検定の場合、WTS と共起に基づく重みを組み合わせることは、効果がないことが確かめられた。

5.4.3 実験 3: 和の結合アプローチによる重み付け

The Wesbury Lab Wikipedia corpus から得られた 10 万文書から各単語ベクトルを学習した。また、単語トピック特定性による重みを求める際のトピックを Gibbs サンプリングに関してはハイパーパラメータ α=0.1 とし、β=0.1 とした。反復数は 2000 とした。単語ベクトルを学習する際のウィンドウサイズを 1 から 10 に変化させたときの和の結合アプローチによって重みを結合させた場合の単語ベクトルのコサイン類似度と各評価データセットの評価値の Spearman の順位相関係数の結果を表 5.4 に示した。なお、和の結合アプローチにおいては、共起に基づく重みは PPMI だけを使用した。コーパスにおけるまた、評価データセット MEN と RW の場合は図 5.6、図 5.7 で表わした。

表 5.5 各手法における Spearman の順序相関係数(和のアプローチ)

weighting	WS	MEN	MTURK	RW	RG
FREQ	0.326	0.360	0.387	0.284	0.335
PPMI	0.398	0.550	**0.518**	0.428	0.444
PPMI+WTS	**0.404**	**0.575**	0.500	**0.434**	**0.453**
FREQ	0.339	0.464	0.516	0.228	0.489
PPMI	0.488	0.651	0.660	0.317	0.550
PPMI+WTS	**0.510**	**0.694**	**0.662**	**0.358**	0.501
FREQ	0.362	0.490	0.574	0.217	0.525
PPMI	0.524	0.673	0.683	0.279	0.514
PPMI+WTS	**0.576**	**0.726**	**0.684**	0.268	**0.550**
FREQ	0.381	0.503	0.606	0.192	0.606
PPMI	0.530	0.680	0.704	0.242	0.624
PPMI+WTS	**0.602**	**0.738**	**0.722**	**0.251**	0.588
FREQ	0.381	0.505	0.605	0.180	0.620
PPMI	0.518	0.679	0.700	0.221	0.594
PPMI+WTS	**0.595**	**0.741**	**0.718**	0.207	0.586
FREQ	0.387	0.505	0.604	0.162	0.645
PPMI	0.512	0.677	0.696	0.193	0.624
PPMI+WTS	**0.601**	**0.741**	**0.720**	**0.207**	0.600
FREQ	0.383	0.508	0.605	0.156	0.654
PPMI	0.505	0.677	0.696	0.188	0.621
PPMI+WTS	**0.603**	**0.743**	**0.720**	**0.204**	0.595
FREQ	0.385	0.508	0.598	0.151	0.650
PPMI	0.516	0.676	0.704	0.183	0.594
PPMI+WTS	**0.609**	**0.744**	**0.720**	**0.207**	**0.638**
FREQ	0.386	0.511	0.599	0.154	0.642
PPMI	0.521	0.675	0.706	0.186	0.585
PPMI+WTS	**0.622**	**0.743**	**0.722**	**0.193**	**0.647**
FREQ	0.381	0.512	0.597	0.151	0.606
PPMI	0.524	0.676	0.708	0.172	0.562
PPMI+WTS	**0.627**	**0.743**	**0.732**	0.162	**0.598**

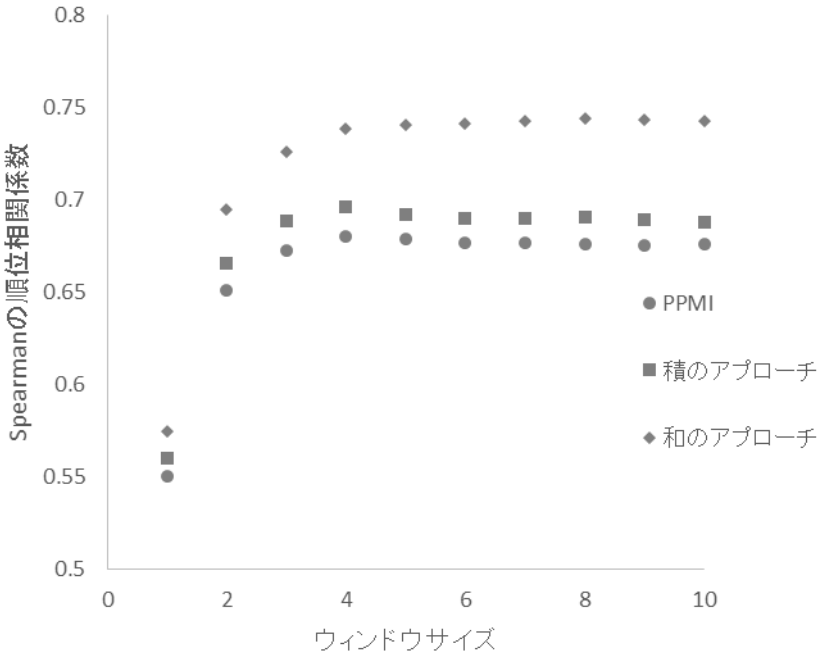

図 5.6 積のアプローチと和のアプローチの比較(MEN)

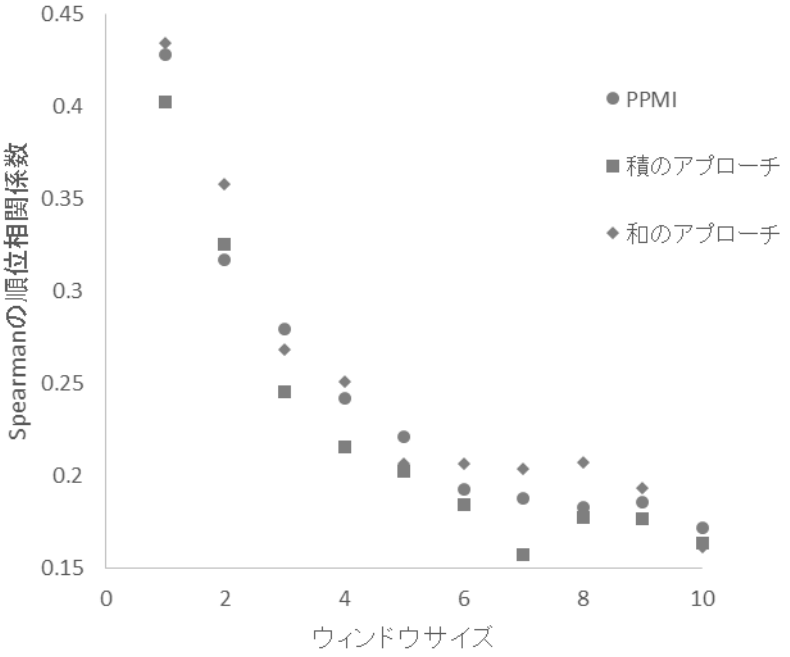

図 5.7 積のアプローチと和のアプローチの比較(RW)

和の結合アプローチによって生成された単語の意味ベクトルは弁別性において、積のアプローチよりもほとんどの場合良いことが分かった。さらに WS や MEN の評価データセットにおいては、共起に基づく重みだけによるベクトルの弁別性よりも平均で 5%以上も改善させている。ウィンドウサイズが 6 のときには、共起に基づく重みだけの場合と和のアプローチによって WTS による重みと組み合わせた場合の Spearman 相関係数の差は 10%近くになる。RW の場合、積の結合アプローチでは、ほとんどの場合において共起に基づく重みだけから生成したベクトルよりも弁別性が悪化するという結果であったが、和のアプローチでは弁別性をほとんどの場合で改善することが確かめられた。

ここまで 2 つのアプローチによる実験を見てきた。次章ではその結果について考察したい。

第６章 議論

前章では、実験の結果をみた。この章では、その実験結果の原因について分析し、考察する。

[1] ウィンドウサイズによる Spearman の順位相関係数の変化

実験の結果、ウィンドウサイズが小さいときは、WTS を考慮した場合と考慮しなかった場合どちらの場合においても単語ベクトルの単語の意味弁別が悪いことが分かった。まず、ウィンドウサイズが小さい場合、十分に単語同士の共起情報を得ることができないからである。意味的に関係している単語まで十分な大きさのウィンドウがない。例えば以下の表は Wikipedia のコーパスにおいて同じ品詞が出現するまでの距離の期待値である。

表 6.1 同じ品詞が出現するまでの距離の期待値

品詞	距離
名詞	3.2
形容詞	7.1
副詞	9.3
動詞	5.3
助詞	14.3
前置詞	6.1
冠詞	7.2

表 6.1 をみると、同じ品詞が出現するまで 5 単語以上の距離を要することが分かる。ウィンドウサイズが小さいと、同じ品詞を持つ単語同士の共起を数えることができない。名詞との共起によってその単語のドメイン性つまり意味を知ることができると Turney は記述している[23]が、目標単語が名詞でありウィンドウの大きさが小さいならば、名詞との共起を考慮することができず、単語の意味的な性質を捉えることができない。また、ウィンドウサイズが小さければ、特定の品詞との共起しか数えることができない。右隣、左隣にある単語の品詞は、多くの場合において同じ品詞の単語である。したがって、ウィンドウサイズが小

さいと、そのような特定の品詞の単語との共起しか数えられなくなり、単語のベクトルにおいてより大きな重みを与えられる文脈単語は特定の品詞の文脈単語に偏ったものになるからである。目標単語が名詞の場合、共起に基づく重みが大きなものに機能的な役割を果たす単語が多くなることがあり、評価データセットにおいて、評価される対象のほとんどが名詞であるため、ウィンドウサイズが小さいとき Spearman の順位相関係数が小さかったのであると考えることができる。

次に複数単語からなる句の影響であると考えることができる。ウィンドウサイズが小さいときは、意味的に関係した単語までの共起を数えることができない場合が多いが、関連した単語よりも同じ句を成す単語との共起が比較的多く、共起に基づいた重み付け手法によって目標単語が出現する句と同じ句に出現する共起単語によりに重みが与えられるからである。表 6.1 を見てもわかるとおり、冠詞が現れてから次の冠詞が現れるまでの距離の期待値は 7 であるにも関わらず、名詞が現れてから次の名詞が現れるまでの距離の期待値は 3 である。つまり、一定数の名詞句の存在が暗示されている。実際、図 6.1 のように句を構成する単語同士の PPMI の値は大きい傾向にある。図 6.1 においての PPMI 値はそれぞれ second-word、rural-area、power-plant、carbon-dioxide、natural-gas の単語対によるものである。それぞれの単語対は句”second world war”、”rural area”、”power plant”、”carbon dioxide”、”natural gas”を成す単語から形成されている。なお、以下の図における縦軸の比率とは、ウィンドウサイズが 1 であるとき、右側の単語を目標単語とした場合の最大の PMI 値に対する単語対の PPMI 値の比率を 1 とした場合の各ウィンドウサイズにおける比率のことである。

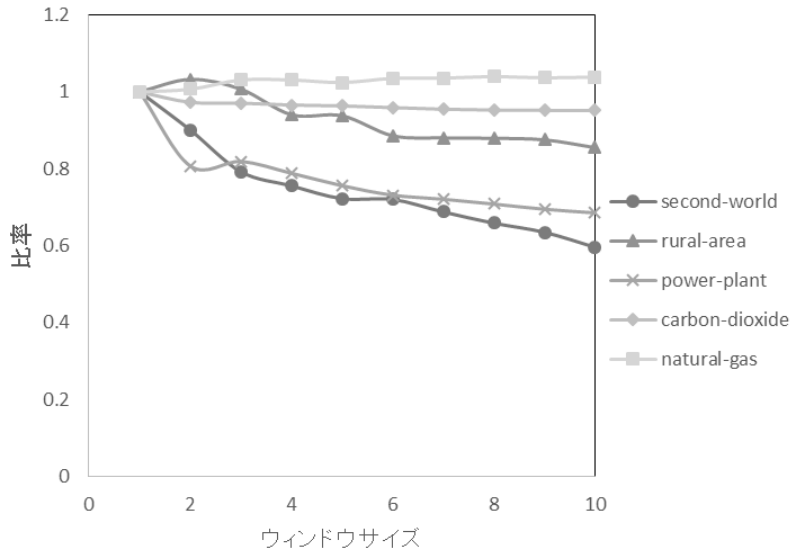

図 6.1 名詞句を形成する単語同士による重みの比

句を成す単語同士でも PPMI の重みの変化が違うことが図 6.1 からわかる。例えば、natural-gas の PPMI 値はウィンドウサイズが大きくなると大きくなるが、second-world の PPMI 値はウィンドウサイズが大きくなると小さくなる。しかし、ほとんどの場合においてウィンドウサイズが 1 の場合より、PPMI の最大値に対する比率は小さくなる傾向にある。このような句によって共起に基づく重みがゆがめられるために、ウィンドウサイズが小さいとき、ベクトルの弁別性が悪くなるのである。

ウィンドウサイズが大きくなるにつれて、WTS を考慮した場合と考慮しなかった場合どちらの場合においても単語ベクトルの単語の意味弁別は安定してくる。これは、文脈ウィンドウ内で共起する単語が増え、単語ベクトルが密になるからである。文脈ウィンドウが最大に達した後、単語ベクトルの弁別はあまり悪化しないが、これはたとえ関係ない単語と共起してしまっていても、目標単語と遠い位置にある共起単語は、偶然共起したものであるから共起に基づく重み手法によって小さな重みしか与えられないからであると考えることができる。

[2] WTS を考慮したときの精度の向上と低下(PPMI)

実験の結果、WTS を考慮した場合の重み付け手法によるベクトルの単語の意味弁別は、WTS を考慮しなかった場合と比べて、ウィンドウサイズが 1 のとき以外は改善した。またウィンドウサイズが 5 程度になるとき最大の改善率を見せた。

この理由の一つとして、単語が出現する文型に影響されていることが考えられる。ウィンドウサイズが 1 のとき、WTS を考慮した重み付けによる単語ベクトルの意味弁別は低下してしまった。この理由としては、共起の必然性を持つ単語が意味的な関連のない単語となることにある。ウィンドウサイズが 1 である場合、PPMI の値が大きくなる共起単語、つまり共起の必然性を持った共起単語は、主に目標単語と句を成す単語であるということを前述したが、句を成す単語同士は意味的関連性のないにも関わらず、共起に基づく重みは大きい。また、その WTS を融合する際、共起に基づく重みも大きいにかかわらず、さらにその句が名詞句である場合、その構成単語である名詞により大きな重みを与えてしまうことになる。名詞は WTS において具体性のある単語として評価されることが多いからである。例えば、単語トピック特定性の値を範囲で分けての品詞の分布は以下のようになる。

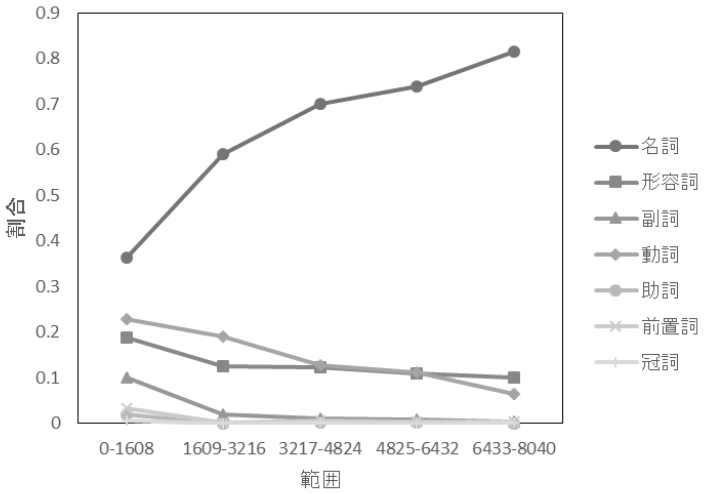

図 6.2 単語トピック特定性に対する品詞の分布

図 6.2 をみればわかるように、名詞はどの WTS の値域においても最も多く分布しているが、高値域では品詞における比率は 8 割以上に達する。形容詞などは、WTS の低値域ではある程度の割合を占めているが、高値域になるとほとんど分布しなくなる。つまり、WTS の値が大きい単語はほぼ名詞であるといえる。したがって、目標単語を構成要素とする名詞句が存在する場合、その共起単語に対する重みは WTS に基づく重みも共起に基づく重みも非常に大きくなっているため単語ベクトルの重みを不当に大きくしてしまうために、ウィンドウサイズが小さいとき WTS との結合が上手くいかなかったのであると考えることができる。

ウィンドウサイズが上がるとともに、句を形成する名詞との共起性による重みは低下する。これによってウィンドウサイズがある程度大きくなった場合は名詞句などが存在していても共起に基づく重みは小さくなるため、たとえ WTS が大きかったとしても相対的に非常に大きな重みにはならない。共起に基づく重みだけの場合は、ウィンドウサイズがある程度の大きさになった後、単語ベクトルの弁別性が安定することを述べた。WTS を考慮した場合もウィンドウサイズがある程度の大きさになった後、単語ベクトルの弁別性が安定する。しかし、ウィンドウサイズが 5 程度になったときに共起性のみの重みによるベクトルによる Spearman 相関係数と WTS も考慮したベクトルによる Spearman 相関係数の差が最大になったあと、ウィンドウのサイズが大きくなるに従って徐々にその差は狭まっていく。これは、WTS が共起に基づいた重みを不当に引き上げてしまうからである。たとえ、共起に基づく重みが小さかったとしてもその値が 0 以上であれば、その小さな重みも WTS によって大きくされてしまうのである。

[3] t 検定における WTS との結合での精度があまり向上しなかった理由

共起に基づく重み付けが t 検定の場合、Spearman の順位相関係数は共起に基づく重みが PPMI の場合と比較してあまり改善しなかった。これは、t 検定と PPMI の重みの分布の違いに起因するものであると考えられる。例えば、単

語”scientist”による各単語の重み分布は降順で図 6.3 のようになる。なお、図における重みは 1 万文書で学習した場合とし、縦軸を重みの最大値に対する各文脈単語に対する重みの比とする。

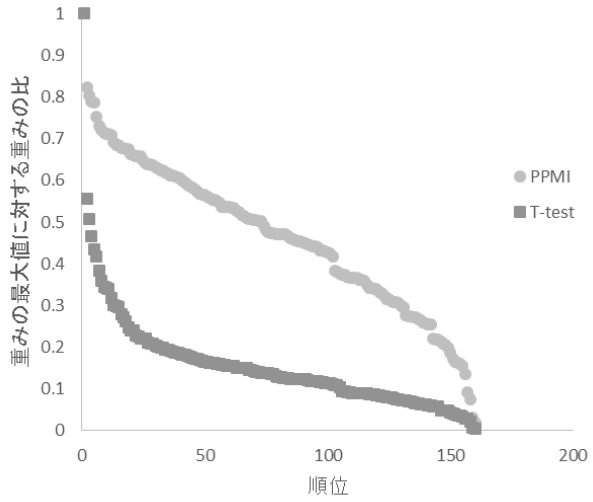

図 6.3 文脈単語の重みの分布の違い("scientist")

図 6.2 を見ればわかるように、t 検定の重みは一番共起の必然性があるものにほとんどの重みを与え、他の大半の重みは最大の重みと比較してとても小さい。一方、PPMI には、与えられた重みの分布に極端な歪みは見られない。t 検定は極端に上位の文脈単語に重みが集中してしまっている。このため、WTS の重みと組み合わせたところで、文脈単語の重みの順序は変わらず、共起による重みをゆがめるだけでとどまってしまうからである。

[4] 和のアプローチによるベクトルの弁別性が良い理由

和のアプローチによる方法は、文脈選択の効果を含んでいるから単語ベクトルの弁別性を良くすることができたと考えることができる。ほとんどの共起単語において抽象的な単語は目標単語に対する共起単語の PPMI を降順に並べた際、下位に存在する。よって、それらの単語が WTS の対数を取ったものを加えることによって、0 になる。一方、積のアプローチでは PMI 値が正であった場合、いくら WTS 値が小さかったとしても 0 になることはない。ある程度の重みが残ってしまう。これゆえ、和によるアプローチの方が、Spearman 相関係数が大幅に大きくなったのだと考えることができる。

さらに以上の理由により RW の評価データセットにおいても Spearman の相関係数が良かったと考えることができる。

第 7 章 結論

7.1 まとめ

本研究は、単語トピック特定性によって、特定のトピックでしか用いられない単語により大きな重みを与えることで、既存の重み付け手法においての共起性に対する偏重を軽減する手法を開発することを目的としていた。潜在 Dirichlet 配分法を用いて、各単語に対するトピックの確率分布と典型的な抽象的単語の単語に対するトピックの確率分布であるトピックの確率分布の距離を KL ダイバージェンスによって計算し、それを単語トピック特定性と定義した。その値を PPMI や t 検定によって計算される共起性に基づく重みと積のアプローチと和のアプローチによって結合させ、より質の良い重み付けを行えるように試みた。

Wikipedia から無作為に選択した 10 万文書を用いて実験を行ったところ、共起だけに基づいた重み付けが PPMI である場合、積による結合アプローチ、和による結合アプローチ、どちらの場合においても提案手法は単語ベクトルの弁別性を改善させることができた。特に和による結合アプローチにおいては、最大で 10%近くも Spearman の順位相関係数を改善させており、自動的に言語資源を作る際に対象となるであろう、特殊語に関してもベクトルの弁別性を向上させることができた。しかし、残念ながら t 検定の重み付けによる場合では、提案手法によるベクトルと既存手法によるベクトルはほとんど変わらなかった。これは分布の歪みで適切な重みを共起に重みに対して加えることができなかったからである。

このように t 検定による重み付けを改善することはできなかったが、PPMI に基づく重み付けは大幅に改善することができた。よって本研究の目的はすべてではないが、部分的には達成できたといっても良いだろう。

7.2 今後の展望

本実験では共起に基づく重みが PPMI である場合、提案手法によってより弁別性の高いベクトルを生成できることを確認した。しかし、t 検定のような非常に重みの分布が歪んでいる場合は、あまり良い重み付けができないことが分かっ

た。今後は共起による重みの分布の尖度や歪度に応じて加える WTS による重みを変動させるような学習法を作らなければならない。そのためには教師なしアプローチによる手法ではなく、教師付きアプローチによる手法を開発しなければならない。

また、この研究において提案した単語トピック特定性は、単語ベクトルの生成だけでなく、文書分類や検索システムなど様々な自然言語処理分野の応用において活用することができるだろう。例えば、単語トピック特定性を初歩的な知識しかないユーザーには、簡単な文書が上位に来るように、専門的な知識があるユーザーに対してはより専門的な文書を上位に表示するような検索システムに用いることができる。ユーザーの知識を、クエリを構成する単語トピック特定性から推定し、その推定に応じて各文書における単語の重みを変動させれば、良いだろう。

謝辞

本研究を進めるにあたり、約 2 年間の間ご指導いただいた北陸先端科学技術大学知識科学研究科知識メディア領域の Ho Tu Bao 教授、LE Tu Ngoc 助教授に感謝いたします。また、各自の研究で忙しい中、本研究の内容などについて時間を割いて議論してくださり、アドバイスを頂いたホー研究室の皆様にも厚く感謝いたします。

最後に、日ごろから応援してくださった、友人や両親に心より感謝いたします。

参考文献

[1] Sahlgren, Magnus. "The Word-Space Model: Using distributional analysis to represent syntagmatic and paradigmatic relations between words in high-dimensional vector spaces." (2006).
[2] Turney, Peter D., and Patrick Pantel. "From frequency to meaning: Vector space models of semantics." Journal of artificial intelligence research 37.1 (2010): 141-188.
[3] Clark, Stephen. "Vector space models of lexical meaning." Handbook of Contemporary Semantics, Wiley-Blackwell, à paraître (2012).
[4] Curran, James Richard. "From distributional to semantic similarity." (2004).
[5] Harris, Zellig S. Distributional structure. Springer Netherlands (1970).
[6] Firth, John Rupert. Papers in Linguistics 1934-1951: Repr. Oxford University Press (1961).
[7] Jurafsky, Dan, and James H. Martin. Speech and language processing. Pearson (2014).
[8] Bullinaria, John A., and Joseph P. Levy. "Extracting semantic representations from word co-occurrence statistics: A computational study." Behavior research methods 39.3 (2007): 510-526.
[9] Bouma, Gerlof. "Normalized (pointwise) mutual information in collocation extraction." Proceedings of GSCL (2009): 31-40.
[10] Scheible, Silke, Sabine Schulte Im Walde, and Sylvia Springorum. "Uncovering Distributional Differences between Synonyms and Antonyms in a Word Space Model." IJCNLP. 2013.
[11] Bordag, Stefan. "A comparison of co-occurrence and similarity measures as simulations of context." Computational linguistics and intelligent text processing. Springer Berlin Heidelberg, 2008. 52-63.
[12] Blei, David M., Andrew Y. Ng, and Michael I. Jordan. "Latent dirichlet allocation." the Journal of machine Learning research 3 (2003): 993-1022.
[13] Blei, David M., and John D. Lafferty. "Topic models." Text mining: classification, clustering, and applications 10.71 (2009): 34.
[14]奥村学、佐藤一誠　『トピックモデルによる統計的潜在意味解析』　コロナ社 (2013)
[15] Griffiths, Thomas L., and Mark Steyvers. "Finding scientific topics." Proceedings of the National Academy of Sciences 101.suppl 1 (2004): 5228-5235.
[16] Griffiths, Tom. "Gibbs sampling in the generative model of latent dirichlet allocation." (2002).
[17] Baker, Kirk. "Singular value decomposition tutorial." The Ohio State University 2005 (2005): 1-24.
[18] Golub and Van Loan, Matrix Computations, Third edition. Johns Hopkins Univ. Press (1996).
[19] Arun, R., et al. "On finding the natural number of topics with latent dirichlet allocation: Some observations." Advances in Knowledge Discovery and Data Mining. Springer Berlin Heidelberg (2010). 391-402.
[20] Teh, Yee Whye, et al. "Hierarchical dirichlet processes." Journal of the american statistical association (2012).
[21] Kullback, Solomon. Information theory and statistics. Courier Corporation (1968).
[22] Cover, Thomas M., and Joy A. Thomas. Elements of information theory. John Wiley & Sons (2012).
[22] Lin, Jianhua. "Divergence measures based on the Shannon entropy." Information Theory, IEEE Transactions on 37.1 (1991): 145-151.
[23] Turney, Peter D. "Domain and function: A dual-space model of semantic relations and compositions." Journal of Artificial Intelligence Research (2012): 533-585.
[24]http://www.psych.ualberta.ca/~westburylab/downloads/westburylab.wikicorp.download.html
[25] Finkelstein, Lev, et al. "Placing search in context: The concept revisited." Proceedings of the 10th international conference on World Wide Web. ACM (2001).
[26] Bruni, Elia, et al. "Distributional semantics in technicolor." Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers-Volume 1. Association for Computational Linguistics (2012).
[27] Radinsky, Kira, et al. "A word at a time: computing word relatedness using temporal semantic analysis." Proceedings of the 20th international conference on World wide web. ACM (2011).
[28] Luong, Minh-Thang, Richard Socher, and Christopher D. Manning. "Better word representations

with recursive neural networks for morphology." CoNLL-2013 104 (2013).
[29] Rubenstein, Herbert, and John B. Goodenough. "Contextual correlates of synonymy." Communications of the ACM 8.10 (1965): 627-633.
[30] Zar, Jerrold H. "Spearman rank correlation." Encyclopedia of Biostatistics (1998).
[31] McDonald, John H. Handbook of biological statistics. Vol. 2. Baltimore, MD: Sparky House Publishing,(2009).
[32] Matsuo, Yutaka, and Mitsuru Ishizuka. "Keyword extraction from a single document using word co-occurrence statistical information." International Journal on Artificial Intelligence Tools 13.01 (2004): 157-169.
[33] 福原知宏, 武田英明, and 西田豊明. "統計情報と概念知識を用いたテキスト間の話題特定." 情報処理学会研究報告知能と複雑系 (ICS) 1999.1 (1999): 1-8.

発表論文

[1]中山雄貴、Ho Tu Bao　単語トピック特定性を用いた文脈単語の重み付け　言語処理学会第 22 回年次大会(NLP2016)　2016 年