修士論文

潜在的意味解析による
中国語のインターネット新語に関する研究

（A Study to Unravel the Characteristic of Chinese Internet New Words Using Latent Semantic Analysis）

那　小川

東京大学大学院　情報理工学系研究科　電子情報学専攻

指導教員　安達　淳

2012 年 2 月 8 日提出

概要

近年、インターネットの迅速な発展に伴い、中国のインターネットユーザ数が 2011 年の 6 月までに 4.85 億人に達していた[1]。今では中国最大の情報源として、インターネットが生活に欠かせない存在になっている。そんな中で、毎日新語があらゆるコミュニティによって作られている。去年では政府が認めるだけでも 500 語の新語が誕生した。また、毎年今年の流行語ランキングが国営テレビによって発表されるなど、これらの新語がインターネットにとどまるわけもなく、普段の生活にも驚異的なスピードで浸透してきている。これは日本語や英語には決して見られない現象であり、現在ではインターネットの新語を知らない人が中国の 20 代とのコミュニケーションが取れないほどにまでなったのである。

言語学者たちの多くもこのインターネット新語に注目し、その由来や適用範囲について言語学的な側面で研究をし始めている。社会心理学にとっても重要な研究のテーマに数えられる。しかしながら、インターネットの新語にはコミュニティー特有の要素も含まれるため、理解するのに必要な前提知識を共有せずには十分に理解できないのが現状である。また、人間によって新語をインターネットから収集、分析するには多くの労力を要するもので未だに一部のインターネット新語しか学者たちに理解されていない。なお、人間の分析では客観性に欠くとされることもあり、また大規模のデータには不向きである。テキストマイニングによって、新語のさらなる特徴が発見されることが期待されている。

私の研究ではこの状況を踏まえて、機械的な手法でインターネット新語の分布や、発生、その特徴を調べる手法 ICTCLAS-LSA を提案する。中国語形態素解析の最新の研究成果である ICTCLAS と、人工知能や自然言語処理分野でよく知られている定番の手法である潜在的意味解析をベースに、インターネットの情報だけで中国語のインターネット新語の意味を推測し、その発生分布（コミュニティー）を突き止める。

潜在的意味解析は例えば「主成分分析」と「SVD」など、言語的には一見関係のないような単語に関係付けることができる 1990 年に提案された古典的な情報検索技術である。それは普通の VSM(ベクトルスペースモデル)を元に、単語をドキュメントではなく、コンセプト空間に圧縮し、射影して扱う手法で、近年様々な実験によって人間の認知モデルとの関係性や、その有効性が幅広く知られるようになった。

中国語の新語検出は昔から大きな課題として未だに完全な解決には至っていない。それは、中国が英語と日本語とは違い、単語の明確な境が存在しないからである。中国語の形態素解析には多くの研究がなされ、本研究ではその最も優れた HHMM モデルに基づいて実装 ICTCLAS を採用し、新語の分析のみ行うものとする。もちろん、形態素解析の結果次第では本研究の精度も大きく左右されるものである。

本研究では言語学者などの人間が中国語の新語解析を行う際に使われるツールを開発した。このツールは与えられた新語とコーパスを以て、新語と意味が近い単語群を抽出し、新語が属する単語のクラスタを出力する。言語学者たちはこれらの情報から知らない新語に関するヒントを手早く得られるのである。人間と比べて、本研究では以下のような利点がある。

1.人力では負えないような膨大なデータ解析を行える

2.各々の言語学者の主観に頼らず、客観的なデータ抽出がなされる。

本研究は中国でもっとも人気なネット掲示板から約３万個の HTML ファイルをクローリングし、それらのデータを元に、約 3.8W 個の単語を研究対象にしている。これは LSA の応用では最大規模の実験になっている。実験の結果は主観的な評価になるが、ICTCLAS と LSA によるシステムの有用性が実証されたものだと考えられる。しかし、いくつかの問題点も研究の中で浮き彫りになっており、現状ではまだ実用的な研究とは呼べない。それらの弱点を克服することで、将来は実用的な新語研究補助システムにできることが言える。

目次

第１章

序章

1.1　研究の背景と目的

１９９４年４月２０日、中国でもインターネットへの接続が可能になり、同年５月に中国高エネルギー物理学研究所が中国で始めてのウェブサーバーを立ち上げた。一般ユーザでもインターネットが使えるようになったのは１９９７年である。

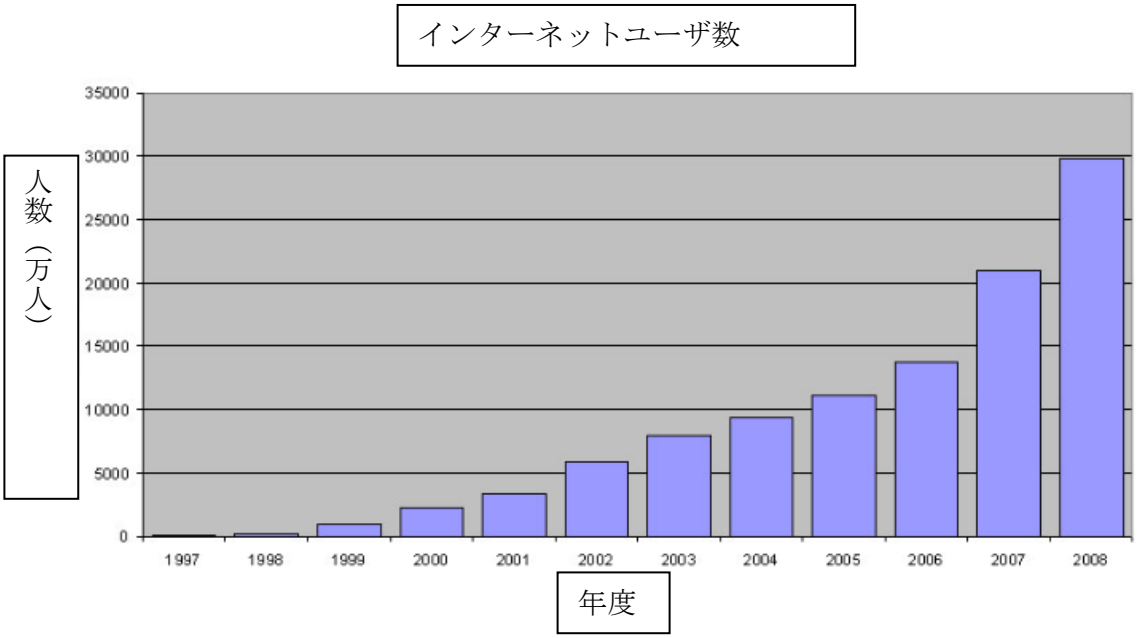

図 1.1　中国のインターネットユーザ数推移

図 1.1 でわかるように、中国でのインターネットユーザが爆発的に増えてきた。2008 年には既に 3.5 億人のユーザがおり、最新のデータでは 2011 年の６月までに 4.85 億人のインターネットユーザが居る[27]。これは人口比で実に約 36％に当たる。日本の約 78％には届かないものの、中国では文化や経済の中心が都市部にあり、またインターネットユーザのほとんども都市部に住まっていることが理由だと考えられる。中国の都市部人口は全人口の約 40％であることから、インターネットユーザ数に見事に一致している。

また、インターネットエコノミーのサイズも GDP の成長率を遥に上回るスピードで大きくなっていっている。これは図 1.2 を見れば一目瞭然である。つまり、インターネットは実体経済や社会に及ぼす影響も日々増大し続け、今では無視してはいけないものになっている。

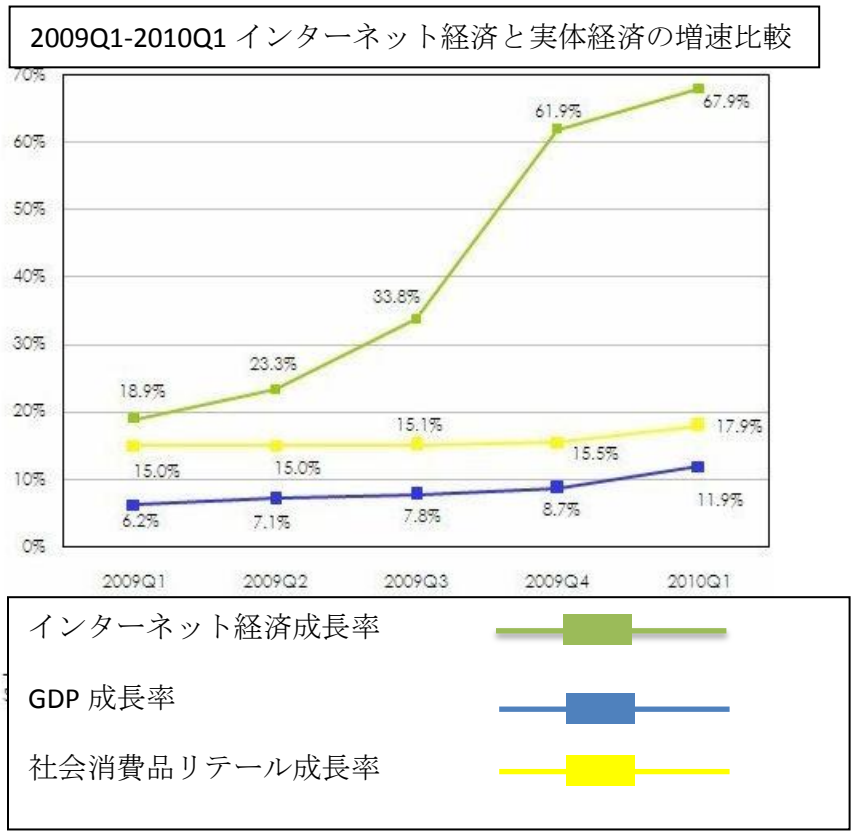

図 1.2 インターネットエコノミーと GDP 成長の比較

図にある青い線は GDP 成長率で、緑色の線はインターネットエコノミーである。

中国の中核は都市部であることから当然ながら、インターネットは現在では中国最大の情報の発生源にもなっている。中国政府のセンサーシップを逃れるように言葉巧みに隠語を作ったり、社会現象を揶揄したりなど、毎日のようにインターネットではそれぞれのコミュニティーによって新語が作られている。さらにこれらの新語はインターネットに限らず、日々の生活にも浸透してきている。大学受験でもインターネット新語を使っている人が急増していることがニュースになるなど、社会的に認知されている。この状況は近年主流メディアにも注目され、毎年「今年の流行語ランキング」も作っている。さる 2010 年では政府が認めるだけでも 500 語以上の新語がインターネットに誕生した。

図 1.3　中国語のインターネット用語が生活にも

また、中国では政府が強いメディア規制を課しており、政府批判の発言によっては投獄される活動家も数多く居る。中国政府は莫大な資金を投じて、[The Great Fire Wall]と呼ばれる政府が管理するセンサーシステムを創り上げた。これは動的に監視する単語リストを保持、更新している。それらの単語は「敏感词」、すなわち「政治的敏感である単語」と一般ユーザに呼ばれ、広く認識されている。

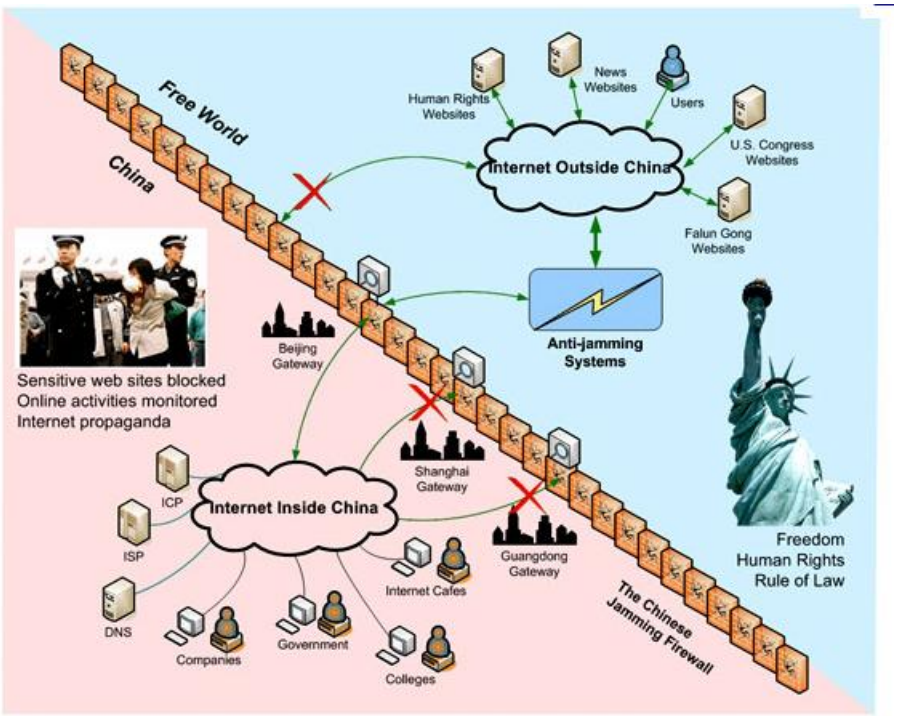

図 1.4　中国の The Great Fire Wall の仕組み

そんな中で、インターネットユーザは政府のセンサーシップを逃れるためにも単語の隠語を作らざるをえない。隠語でも政府に認識されれば、新しい隠語を作るサイクルが延々と続く。

このような状況下で、中国政府主催、大学が参加している研究機関も立ち上がったのである。「国家言語資源観測と研究センター」と名付けられる組織は言語学者とコンピュータ・サイエンスの専門家から構成されている。この研究機関ではインターネット新語に対して様々な面で研究を重ねている。一例を挙げると、社会的なニュースと密接に関わる単語の抽出などが行われていた。

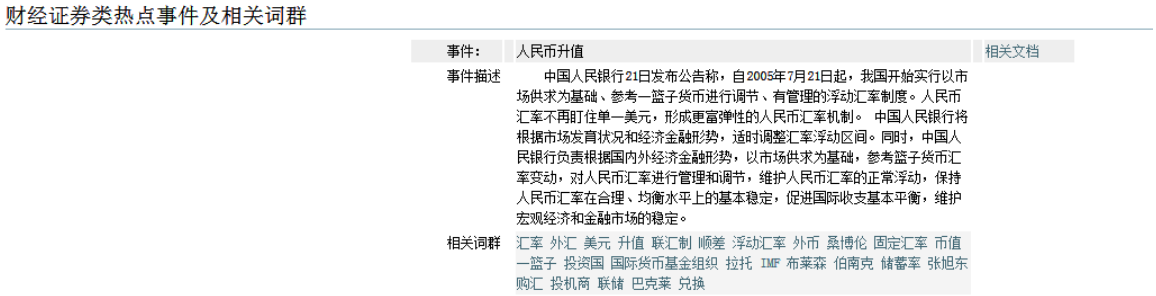

図 1.5　人民元切上げというニュースに関わる単語のクラスタ

言語学的な手段では人間の力が要されるので、大規模なデータの自動処理には不向きである。今まで用いられてきた機械的なアプローチはテキストマイニングであったが、それでは単語の Strict Co-occurrence に制限されてしまう。しかし、図 1.5 にもあるように、人民元切上げに密に関係する単語は人民元切上げと同じ文に現れるとは限らない。本研究では図 1.5 のような用途では潜在的意味解析がより適していると考える[5]。具体的に潜在的意味解析の説明はここでは割愛するが、第 2 章で詳しく述べるものとする。

以上のことにより、本研究では潜在的意味解析をベースに、中国語のインターネット新語の意味解析を近似度計算とクラスタリング手法によって行った。

1.2　本研究の仕組み

インターネット新語を解析するのに採ったアプローチを以下にて図示する。

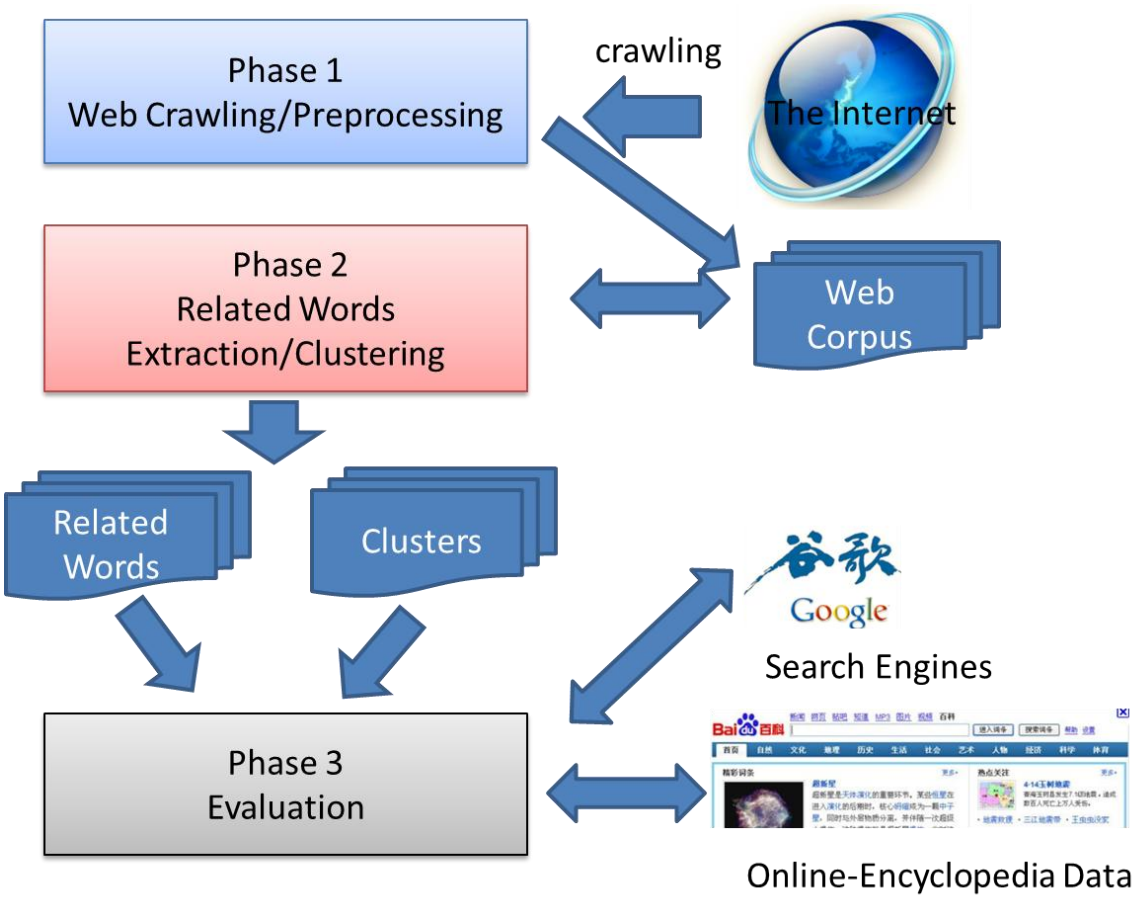

図 1.6　本研究の仕組み

本研究は 3 フェーズに分けている：

1. ウェブクローリング
2. 新語の意味解析
3. 結果評価

この 3 フェーズはそれぞれ図 1.6 に対応しており、中でもフェーズ 2 が本研究で最もコアな部分である。フェーズ 2 の実験に必要な技術は中国語の形態素解析（ICTCLAS）、潜在的意味解析（LSA）及びクラスタリングであり、実験の仕組みを以下の図に示す。

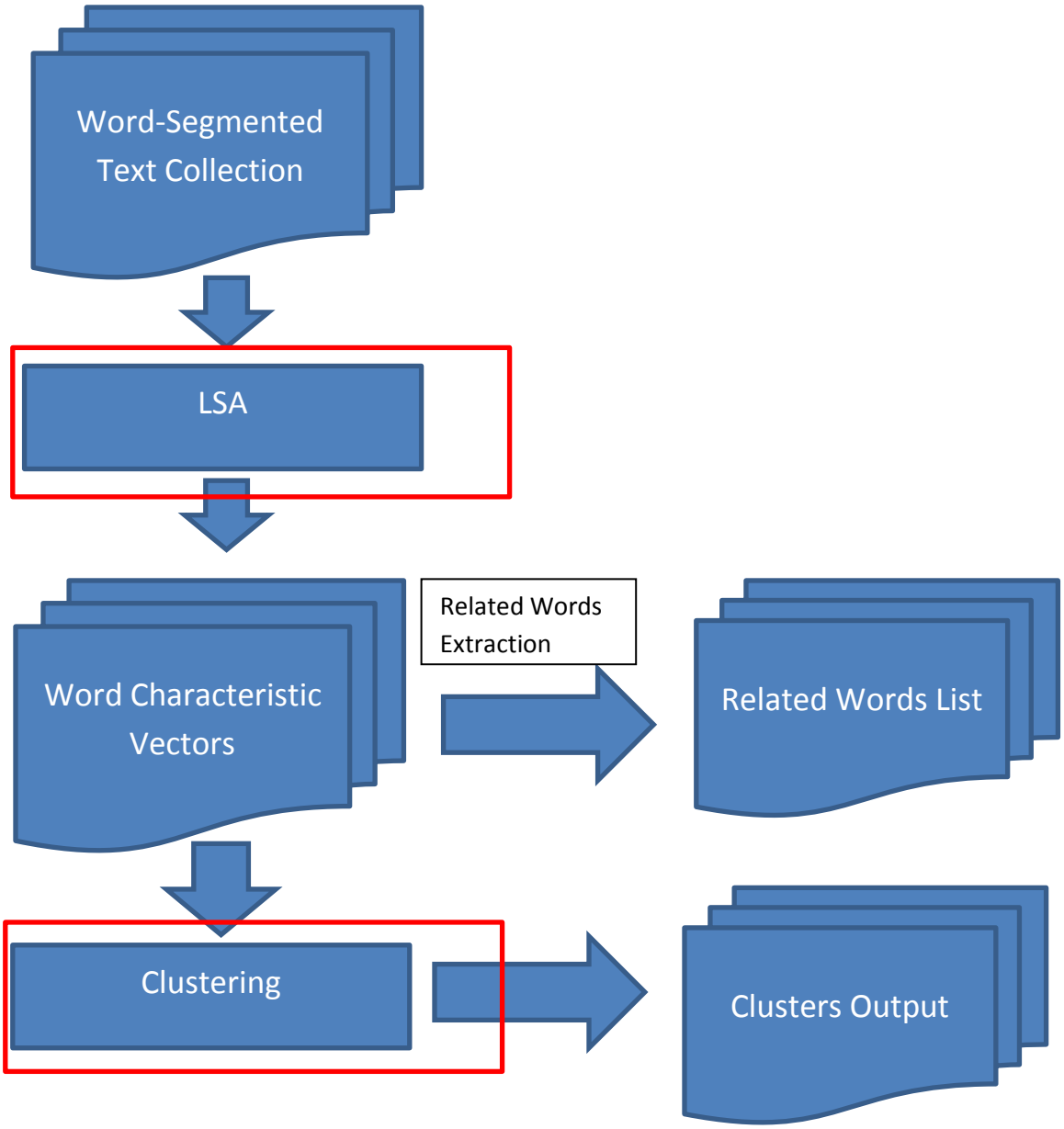

図 1.7　フェーズ 2 実験の手順

図 1.7 に赤く囲まれた部分は関連研究で詳しく述べる。以下に実験のステップを簡単に述べよう。

1.前処理を経たテキスト文書は LSA 処理を行い、単語特徴ベクトルを生成する。

2.1.単語特徴ベクトルを用いて、Similarity 計算による新語の近似語抽出を行う

2.2.単語特徴ベクトルのクラスタリングを行い、クラスタを結果としてアウトプット

1.3　本論文の構成

本論文の構成は以下のとおりである。

第 2 章では研究をする上でコアな技術である中国語の形態素解析と潜在的意味解析及びクラスタリング手法について述べる。

第 3 章では実験のためのデータの用意と前処理について述べる。

第 4 章では中国語のウェブコーパスで行った新語の意味解析、単語のクラスタリングについて詳しく述べるとともに、実験の結果を説明する。

第 5 章は本論文の結論と今後の課題について述べる。

第 2 章

関連研究

2.1　中国語の形態素解析

2.1.1 中国語の特殊性

中国語は現存する世界で最も古い言語にギネスに認定されており、言語学上の分類では、中国語は孤立語になる。古い歴史からもわかるように、中国語はとりわけ複雑である。英語等に比べて、使用される文字の種類が多く、また日本語と同じように単語の自然な境が存在せず（スペースのようなｄｅｌｉｍｉｔｅｒがない）、また単語が格変化や過去形などの変化がないことから、文は単語の組み立てロジックと順序に深く依存している。さらに、古代漢語の遺留である多くの熟語や慣用表現（古代漢語表現）も現代の中国語と似て非なるものなので、一層言語学者たちを苦しめ続けている。

中国語の例文を挙げよう。「今天的太阳好耀眼，我和张小明有说有笑地到金碧辉煌的人民大会堂去参观」。これは「今日は太陽が眩しく、私は張小明と談笑しながら煌びやかな人民大会堂に参観した」という意味である。英語ではこのようになるであろう「I went to visit the people’s parliament with Xiaoming Zhang laughing and talking in the shining sun」。この 3 つの文を比較してみよう。英語は形態素解析（Word Segmentation）が既に完成しており、不要である。ところが、日本語と中国語ではまずそれぞれの単語を認識するどころから始めないといけない。日本語の場合、膠着語であるために、それぞれの単語には必ず文においての成分を表す副詞が付いている。「今日**は**太陽**が**眩し**く**、私**は**張小明**と**談笑**し**ながら煌び**やかな**人民大会堂**に**参観**した**」。これらの副詞に注目すると、単語を切り出すことがそう難しくはなかろう。一方で中国語はかなり難しい。正しく形態素解析すると、このようになるであろう。。「今天**（今日）**的**（副詞、従属関係）**太阳**（太陽）**好**（非常に、副詞）**耀眼**（眩しいという意味の形容詞）**，我**（私）**和**（「と」に相当する副詞）**张小明**（専有名詞）**有说有笑**（談笑の慣用表現）**地**（形容詞を副詞化する際に使われる）**到**（「へ」に相当する意味の副詞）**金碧辉煌**（古来の熟語、意味はおおよそ煌びやか）**的**（副詞、形容詞につく）**人民大会堂**（専有名詞）**去**（「へ」と意味が近い）**参观**（参観）**」。

このようにして、中国語の形態素解析はネイティブスピーカーにはたやすくできるが、かなり難しい技術的な難問になっている。現在のほとんどの自然言語処理（機械翻訳など）は文を単語成分に分解することを要求するので、中国語の自然言語処理も他の言語より扱いが異なってくる。

以上のことで、現在中国語に関する自然言語処理の研究では「形態素解析」にフォーカスしたものがほとんどで、それ以外の研究では人間によって作られたコーパスを使っている。英語のような一般的な言語の自然言語処理の流れは意味の抽出に注目しているが、中国語では完全的な自然言語処理は一般的に大きく２ステップになっている。１．テキストを形態素解析する　２．単語の集合になった文書を必要に応じて処理する

2.1.2 中国語形態素解析の基礎理論

中国語の形態素解析は２０年近く研究されてきたが、未だに活発な研究分野の１つである[15]。そもそも中国語において「単語」とは何か自体も議論の種である。例えば、「人民大会堂」は「人民」と「大」、「会堂」に分けるべきか、それとも１つの専有名詞に見なすべきかは意見が別れるが、ほとんどの単語で概ねの一致が得られている。

中国語の形態素解析はアプローチによって大きく２つに分けられる。１．辞書などの構造的な言語素材に依存する言語学的な手法　２．統計的な手法（辞書が必要）。[14] 辞書を用いる手法でもっとも成功を収めているのは「最長マッチング」手法及びその改良版である。世の中に存在する Wordnet のような言語コーパスをうまく利用している。統計的な手法も辞書を用いるが、文法などに依存せずに、新語をよりよく発見できる。

幸いなことに、中国語には日本語や英語のように単語の格変化や過去形などがないために、単語自体はいつも辞書形である。それを踏まえて、「最長マッチング」手法は文を末尾もしくは先頭からスキャンしていく、辞書に載っている最長のパータンにマッチするシーケンスを「単語」として抽出する。もちろんこれだけでは足りず、中国語の言語的な特徴を生かしたヒューリスティックも多数用いられている。このような手法の問題点としては OOV（out of vocabulary）と呼ばれる辞書に載っている単語にうまく対処できていない点である。本研究ではインターネット新語に注目しているので、なるべく辞書に頼らない方法が望ましく、具体的な手法は本研究では用いないため割愛する。

もう１つの手法も辞書を用いるが、新語発見のために多くの統計的な手法が組み込まれている。本研究ではその中から代表的なものである ICTCLAS を用いている。ICTCLAS は中国科学院計算機研究所が開発した HMM（隠れマルコフモデル）を階層的に応用した形態素解析手法であり、そのパフォーマンスは正確さとスピードの両面で数々のコンテストでトップを獲得している[12]。ICTCLAS については次章で詳しく説明する

2.1.2 HHMM モデルに基づく ICTCLAS

ICTCLAS(Institute of Computing Technology, Chinese Lexical Analysis System)は中国科学院計算機研究所が長年間研究を重ねている中国語の形態素解析プロジェクトである。基本的な設計思想は隠れマルコフモデルを応用し、形態素解析のそれぞれのステップでもっとも確率が高い結果を出力し、最後はそれらの結果を組み合わせて最良のアウトプットを得る。HHMM は(Hierarchical Hidden Markov Model)の略であり、つまり階層的な隠れマルコフモデル[12]。その仕組みを図 2.1 に示す。

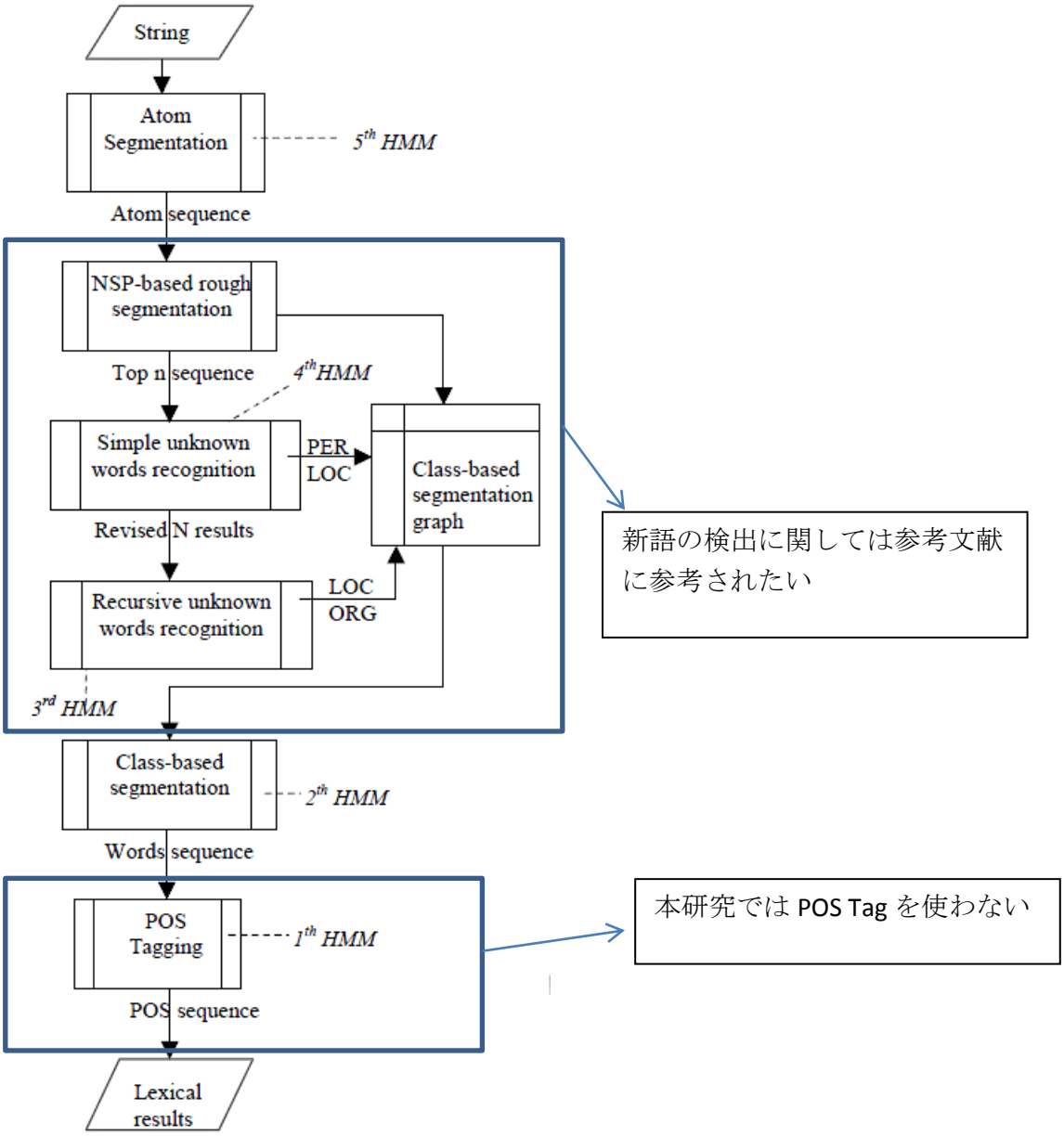

図 2.1.1　ICTCLAS の仕組み

図 2.1 から、全部で 5 段階の HMM が応用されていることがわかる。

まずは文を最小の単位である atom に分割する（atom は句点や漢字 1 文字、アルファベット）。

それから atom をデータとしてステップ 2 に提供する。つまり、パターンマッチングとは違って、ICTCLAS ではそれぞれの atom(文字)を単位に、とある単語に当てはまる最大の確率で形態素解析を行なっている。これを図 2.2 で示す。

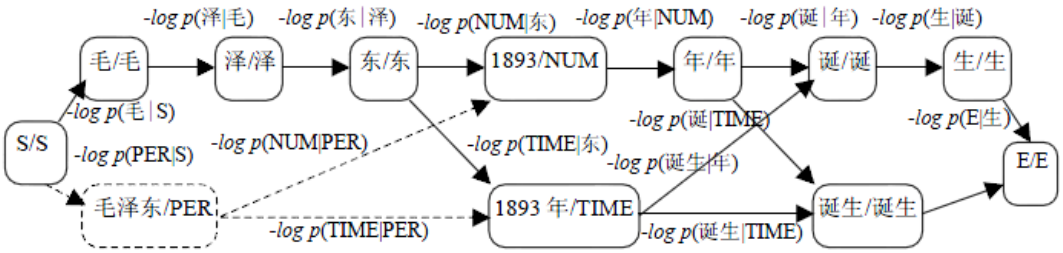

図 2.1.2　ICTCLAS の文字をベースにした HMM モデルの応用

POS タグとは人間が用意するコーパスとほぼ同じで、単語の成分を表したものである。本研究では POS タグを用いていない。

本研究では新語の検出を全て ICTCLAS 手法に委ねており、人間の介入を必要としていない。また、図 2.3 が ICTCLAS の実験結果を示し、その有効性を十分に裏付けている。

Track	ASc	CTBc	CTBo	HKc	PKc	PKo
Participant Number	6	6	7	4	10	8
Corpus Size (bytes)	38,882	125,248	125,248	114,384	56,254	56,254
True Word count	11,985	39,922	39,922	34,955	17,194	17,194
Test Word count	12,360	40,460	40,426	37,274	17,582	17,563
Insertions	434	1,789	1,755	2,439	485	458
Deletions	59	1,251	1,251	120	97	89
Substitutions	506	3,281	3,262	2,291	562	539
Nchange	999	6,321	6,268	4,850	1,144	1,086
Recall (Rank)	0.953 (3)	0.886 (2)	0.887 (4)	0.931 (3)	0.962 (1)	0.963 (1)
Precision (Rank)	0.924 (5)	0.875 (1)	0.876 (4)	0.873 (3)	0.940 (3)	0.943 (2)
F measure (Rank)	0.938 (5)	0.881 (1)	0.881 (4)	0.901 (3)	0.951 (1)	0.953 (2)
OOV rate	0.022	0.181	0.181	0.071	0.069	0.069
OOV Recall (Rank)	0.178 (5)	0.705 (1)	0.707 (5)	0.243 (4)	0.724 (2)	0.743 (2)
IV Recall(Rank)	0.970 (3)	0.927 (5)	0.927 (5)	0.984 (1)	0.979 (2)	0.980 (1)
*Time Cost (s)	3.92	10.57	10.62	7.11	5.18	5.53
**Speed (bytes/s)	9,919	11,849	11,794	16,088	10,860	10,173

図 2.1.3　ICTCLAS の実験結果

2.2　潜在的意味解析

2.2.1 VSM(Vector Space Model)

VSM はベクトルスペースモデル、もしくはベクトル空間モデルと呼ばれ、テキストマイニングや情報検索においては古典的な手法の 1 つに挙げられる[11]。これは、転置インデックスと同じように、まずコーパスに対して単語が文書に出現する頻度を表した行列を作る。図 2.2.1 にその一例を示す。

	s1	s2	s3	s4	s5	s6	s7	s8	s9
human	1	0	0	1	0	0	0	0	0
interface	1	0	1	0	0	0	0	0	0
computer	0	1	0	0	0	0	0	0	0
user	0	1	1	0	1	0	0	0	0
system	0	1	1	2	0	0	0	0	0
response	0	1	0	0	1	0	0	0	0
time	0	1	0	0	1	0	0	0	0
EPS	0	0	1	1	0	0	0	0	0
survey	0	1	0	0	0	0	0	0	1
trees	0	0	0	0	0	1	1	1	0
graphs	0	0	0	0	0	0	1	1	1
minors	0	0	0	0	0	0	0	1	1

図 2.2.1　文書と単語がなす行列の例

図 2.2.1 ではそれぞれの単語が文書に出現した頻度を行列の値にしている。これが最も基本的なベクトルスペースモデルである。こうすることで、単語を文書空間のベクトルで表すことができる。例えば human という単語は 9 次元の空間では(1,0,0,1,0,0,0,0,0)として表されている。

意味が近い、あるいは関係のある単語は同じ文書に現れやすいという考えの元で、文書の数が多ければ、単語と単語の近似度もこの文書空間のベクトルで比較することができる。また、このベクトルスペースモデルでは、クエリを文書に見立てて、文書のベクトルとの比較により文書検索がなされる。

しかしながら、極めて意味のある専有名詞（例えば EPS）が普通の単語(system)と比べると、圧倒的に普通の単語のほうが出現頻度が高くなる。これでは専有名詞などの情報量が多く、出現頻度が相対的に少ない単語を上手に扱えない。この問題を解決するには TF-IDF と呼ばれる頻度ではなく、情報量を行列の値にする手法が提案された。

TF-IDF (Term Frequency- Inverse Document Frequency)

TF-IDF は、文書中の単語に関する重みの一種であり、主に情報検索や文章要約などの分野で利用される。

TF-IDF は、TF（単語の出現頻度）と IDF（逆文書頻度）の二つの指標にもとづいて計算される。[23]

$$\mathrm{tfidf} = \mathrm{tf} \cdot \mathrm{idf}$$

$$\mathrm{tf_{i,j}} = \frac{n_{i,j}}{\sum_k n_{k,j}}$$

$$\mathrm{idf_i} = \log \frac{|D|}{|\{d : d \ni t_i\}|}$$

$n_{i,j}$は単語 i の文書 j における出現回数、｜D｜は総ドキュメント数、$|\{d : d \ni t_i\}|$は単語 i を含むドキュメント数である。

そのため、idf は一種の一般語フィルタとして働き、多くのドキュメントに出現する語（一般的な語）は重要度が下がり、特定のドキュメントにしか出現しない単語の重要度を上げる役割を果たす。

2.2.2 LSA(Latent Semantic Analysis)

LSA は潜在的意味解析と日本語で呼ばれている。1990 年に提案された LSA (Latent Semantic Analysis）[5]は、人手や言語に対する前提知識を一切用いなくとも、シソーラスを利用した場合と同様の結果が得られる手法として知られている。その基本アイディアは検索や文書分類を行う際、直接単語情報を用いずに、その単語情報を単語種類数よりはるかに小さな意味空間に写像した上 で操作を行うというものである。例えば、「bank」という多義性を持った単語は、銀行という意味と、土手という意味がある。この bank の出現頻度を 銀行の意味に 0.6、土手の意味に 0.4 とふりわけて検索すればどちらの意味にも対応できる。LSA では （1）どのような意味の固まりがあるのか（2）各単語がどの意味にどれだけ関係しているのかを自動的に求めることができる。表記上違う単語でも、LSA により同じ意味に写像して扱うことで同義語問題の解決ができる。また、同一の単語であっても複数の意味に写像して扱うことで多義語問題の解決が期待できる。

また、中国語研究の分野ではまだ LSA の応用が近年始まったばかりであり、まだまだこれからの発展が委ねられている。[18][20][21][26]

LSA の仕組み

LSA の仕組みは極めてシンプルである。VSM で作った単語*文書行列を X と置くと、任意の行列 X には SVD 分解が可能なことはわかっている。X を以下のように分解する。
$X = U\Sigma V^T$

さらに、単語と単語の相関関係、文書と文書の相関関係を表す行列は以下である

$$\begin{aligned} XX^T &= (U\Sigma V^T)(U\Sigma V^T)^T = (U\Sigma V^T)(V^{T^T}\Sigma^T U^T) = U\Sigma V^T V\Sigma^T U^T = U\Sigma\Sigma^T U^T \\ X^T X &= (U\Sigma V^T)^T(U\Sigma V^T) = (V^{T^T}\Sigma^T U^T)(U\Sigma V^T) = V\Sigma U^T U\Sigma V^T = V\Sigma^T\Sigma V^T \end{aligned}$$

上の式からは、$U\Sigma$ が単語の特性を表す行列であることがわかる。なぜならば、行列 X の次元を(m,n)と仮定し、つまり m 個の単語と n 個の文書がある。XX^T は(m,m)次元の行列であり、つまり m 個の文書間の対応関係を表している。これが$(U\Sigma)(U\Sigma)^T$である。これは本研究で使われることになる。

これだけであれば、わざわざ SVD 分解した意味がまだはっきりしないが、実は LSA と呼ばれるのは SVD 分解して得た m 次元行列 $U\Sigma$ に対して、次元削減を行う。最大で k(k<=m)次元を残し、(m,k)次元の行列 $U_k\Sigma_k$を得る。行列 $U_k\Sigma_k$の行ベクトルは k 次元ベクトルになり、これは元の TD(Term-Document)行列から得た n 次元の単語ベクトルよりも正確に単語と単語の関係を表していることが実験でわかっている。

つまり、元の TD 行列を SVD 分解し、最大で k 次元残して得た $U_k\Sigma_k$を使うのが LSA なのである。（ここでは文書検索について考慮しない）k は実験によって確定するが、一般的には trace(Σ_k)>=trace(1/2(Σ))を満たす最小の k を取る。

LSA の理論根拠

LSA の理論根拠は以下のように述べられている。

文書には複数の概念（コンセプト）が存在し、単語にも複数の概念に対応していると考える。とある文書の集合 *S*（コーパス）が与えられた時に、この文書の集合には最大で *k* 個のコンセプトが存在し、*m*(*m*>*k*)個の単語が文書にある。SVD 分解して得た $U_k\Sigma_k$ はそれぞれの単語が *k* 個のコンセプトにどれくらい関係しているかを表している。以下に一例を挙げるよ。

図 2.2.1 は図 2.2.2 のような明確に違う 2 つのコンセプトを持つ文書集合から作られた

c1: *Human* machine *interface* for ABC *computer* applications
c2: A *survey* of *user* opinion of *computer system response time*
c3: The *EPS user interface* management *system*
c4: *System* and *human system* engineering testing of *EPS*
c5: Relation of *user* perceived *response time* to error measurement

m1: The generation of random, binary, ordered *trees*
m2: The intersection *graph* of paths in *trees*
m3: *Graph minors* IV: Widths of *trees* and well-quasi-ordering
m4: *Graph minors*: A *survey*

図 2.2.2　2 つのコンセプトを持つ文書集合（コーパス）

このコーパスでは、コンセプトの数を k=2 に設定して LSA 分解を行うと、図 2.2.3 のような結果が得られた。

human	0.7	-0.3
interface	0.7	-0.2
computer	0.8	0.1
user	1.3	0.2
system	2.1	-0.4
response	0.9	0.3
time	0.9	0.3
EPS	1.0	-0.4
survey	0.7	0.7
trees	0.0	1.2
graphs	0.1	1.6
minors	0.1	1.1

C1-C5 のコンセプトに強く関係する単語は次元 1 の値が大きい

M1-M4 のコンセプトに強く関係する単語は次元 2 の値が大きい

図 2.2.3　LSA 分解後に得た $U_k\Sigma_k$(k=2)

以上の例では、LSA の有効性が示せたと考えられる。実際のデータはサイズが大きいとこのように人間によって k を決めるわけにはいかず、実験を行うなどして k を決めるケースが多い。

また、近年には PLSA と呼ばれる手法が提案され、LSA よりも精度がいいという結果が報告されている。その理由は LSA は単語の分布が正規分布であると暗黙的に仮定しているが、実際の単語の確立分布を考慮した Probabilistic LSA が精度がよりよい[24]

まとめると、LSA とは SVD 分解と次元削減によって、単語の相関関係のノイズを除去し、本当の関係をより強くすることで VSM の精度を大幅に改良した手法である。次節で LSA の応用や意義について詳しく述べる。

2.2.3 LSA の意義と応用

LSA においてコンセプトの概念が近年心理学者たちの研究で、人間の言語や知識の獲得にとても近いモデルだったことがわかった[21][25][26]。情報科学の分野でも多くの科学者が LSA をいろんなテーマに応用している。言語学では中国語の漢字の多義性の解消や[21]、低学年の生徒の作文自動採点など内容は様々であった。

LSA を応用するには学習コーパスというものが必要不可欠で、その学習コーパスの質と量によって LSA の結果は大きく変わってくる。とある研究では高校生までに読むであろう文書を LSA に学習させて、TOEFL(非ネイティブのための英語力試験)の同義語問題を解かせてみたどころ、TOEFL 受験者の平均点とほぼ同じ結果になった[9]。なお、LSA の根拠は単語間のコンセプトであるために、同義語と反義語はほぼ同じ得点になることが予想され、また実際に確認されている。これは LSA にとって大きな欠点ではあるが、今回の実験では「同義語」よりもむしろ新語の性質を知ることが大きい。

つまり、LSA は完璧無欠な理論ではなく、「人間性」と似たものが入っていると考えられる。そのためか、近年では自然言語処理においては言語学のヒューリスティックに劣ると思われている。筆者はそれがコーパスの質と量によるどころが大きいではないかと思っている。

典型的な実験で LSA にかける学習コーパスは数千程度であり、この程度の文書では当然人間の言語学エキスパートが作るヒューリスティックルールに負けるであろう。SVD 計算はとても時間がかかるもので、計算パワーやメモリの制限で、LSA の有用性が制限されてしまうが、人間の知識を要しないどころがユニークで大きいと感じる。

LSA の次元削減

LSA の最適な次元を決める方法として 1 つの例を図 2.3.1 に挙げる。

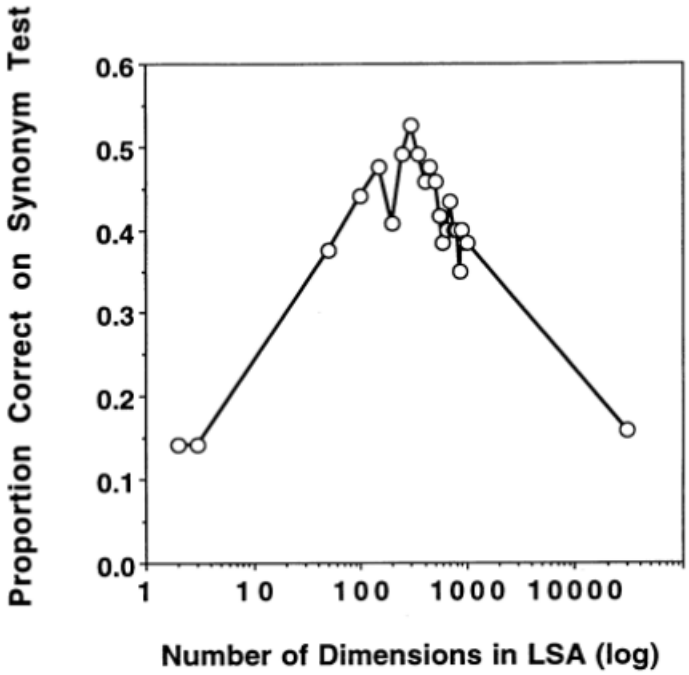

図 2.3.1　同義語テストを用いた最適次元の発見

図 2.3.1 に示したのは LSA に 30473 個の段落を学習させ、6 万個の単語で得た結果を TOEFL の同義語テストで得た点数のグラフである。LSA が 500 次元の際に、一番高い点数が得られた[9]。

もちろん、これだけで LSA の最適な次元が 500 と断言するのはできないが、LSA にとって次元数が多すぎても、少なさすぎてもよくないことが言えるだろう。

LSA が文書検索においての応用

本研究とは直接に関係しないが、LSA のもう 1 つ大きな応用は情報検索である。クエリとして与えられた文を 1 つの文書とみなし、それに含まれる単語のベクトルから構成され、コンセプト空間にある文書ベクトルと比較する。

手順はまず全ての単語ベクトルを足しあわせて、クエリ単語ベクトル q を作る。それから以下の式に従ってコンセプト空間の文書クエリ q^を作る。

$$\hat{\mathbf{q}} = \Sigma_k^{-1} U_k^T \mathbf{q}$$

次は文書クエリとコンセプト空間にある他の文書との近似度を計算して、閾値を超える文書を答えとして返せばいい。

このような情報検索システムのメリットの 1 つは精度の高い情報検索ができることである。つまり厳密なテキストマッチングではなく、セマンティック（意味）で検索できることである。「SVD」と入力して、「主成分分析」が書いてある文書が帰ってくる。

もう 1 つのメリットとしては言語横断の検索ができるのである。同じ文書の英語版とフランス語版を用意し、TD 行列では行列の対応関係をしっかり確定すれば、フランス語で英語の文書を正しく取り出せる。

但し、SVD には多くの時間がかかるため、リアルタイムの応用には不向きである。簡単にコーパスを追加することができず、またウェブデータのような大きなコーパスは処理不可能である。そのため、現在のサーチエンジンでは検索技術として採用されていない。

2.3　クラスタリング手法

クラスタリングは情報検索やデータマイニング分野で古くから知られている手法であり、データ間の関係性や分布特徴を調べるのによく使われてきた[2][19]。メトリック空間に分布している点が一定の区域に集中している現象があり、その集中したデータ群をクラスタと呼ぶ場合がある。

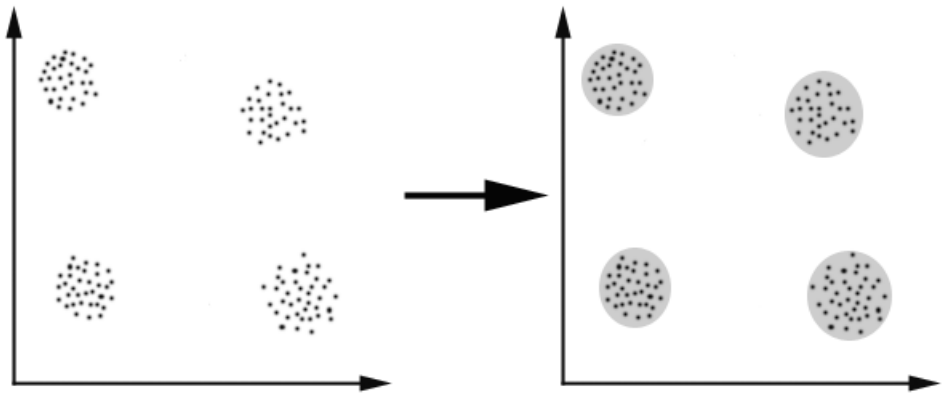

図 2.3.1　空間に分布している点が４つのクラスタをなす例

つまり、クラスタリングの根拠はデータ（点）が密に分布していることに着目し、それらの点に共通性を見出す手法である。しかしながら、データの分布特性によって、クラスタを厳密に定義することができない。許す距離の閾値によっては上の例では２つのクラスタをなしていると言うこともできるだろう。

最終的にクラスタをどのように見つけるのかは２つの要因に関係している。

① 距離の計算（データの関係性をどう定義するのか）
② クラスタの定義（何を以てクラスタとするのか）

メトリック空間における距離の計算は様々な方法で得られるが、以下に述べる公理を満たす必要がある[3]。

オブジェクト集合D と距離関数$d : D \times D \rightarrow \mathrm{R}$ から成る空間$M = (D, d)$ が,
non-negativity: $\forall x, y \in D, d(x, y) >= 0$
symmetry: $\forall x, y \in D, d(x, y) = d(y, x)$
identity: $\forall x, y \in D, x = y \leftrightarrow d(x, y) = 0$
triangle inequality: $\forall x, y, z \in D, d(x, z) <= d(x, y) + d(y, z)$
の 4 つの公理を満たすとき，メトリック空間という.

このような公理を満たす距離は例えばコサイン距離、ユークリッド距離、ミンコフスキー距離、マンハッタン距離などがよく使われる[7]。

クラスタの定義は通常極めて曖昧になりがちである。例えば、１つのデータは１つのクラスタに属するのか、あるいはいくつかのクラスタにある程度属するのか。クラスタ数は与えられているものとするのか、それとも閾値を用いて判断するのかなど、実に複雑である。本研究での応用を考慮し、ここでは１つのデータが１つのクラスタに属する前提とする。

このようにすると、クラスタリング手法は２つに分けることができる。

① パラメータ付き（クラスタ数が予め決まっている）
② パラメータなし（クラスタ数が計算の結果で決まる）

パラメータ付きの手法では主にとあるデータセットに対し、様々な前提条件を決め、これらの条件を満たす最適な解を求めるものである。
一方で、パラメータなしの手法ではデータから再帰的にクラスタを形成（もしくは分割）していく。決めた閾値に達するまでプロセスを繰り返し、最終的にクラスタを得る。
以下では本研究で用いられる代表的なパラメータ付きの手法 K-means とパラメータなしの手法 Hierarchical Clustering について紹介する。

K-means

K-means はクラスタリング手法の中でも一番古いアルゴリズムとなる。その主旨は以下のようである。

K-平均法は、一般には以下のような流れで実装される。データの数を n 、クラスタの数を K としておく。

1. 各データ$x_i(i = 1 \cdots n)$に対してランダムにクラスタを割り振る。
2. 割り振ったデータをもとに各クラスタの中心$V_j(j = 1 \cdots K)$を計算する。計算は通常割り当てられたデータの各要素の平均が使用される。
3. 各 x_i と各 V_j との距離を求め、x_i を最も近い中心のクラスタに割り当て直す。
4. 上記の処理で全ての x_i のクラスタの割り当てが変化しなかった場合は処理を終了する。それ以外の場合は新しく割り振られたクラスタから V_j を再計算して上記の処理を繰り返す。

結果は、最初のクラスタのランダムな割り振りに大きく依存することが知られており、1 回の結果で最良のものが得られるとは限らない。また、この手法では、強制的に K 個のクラスタにするため、大きいクラスタを分割して 2 つとして認識するケースもある。

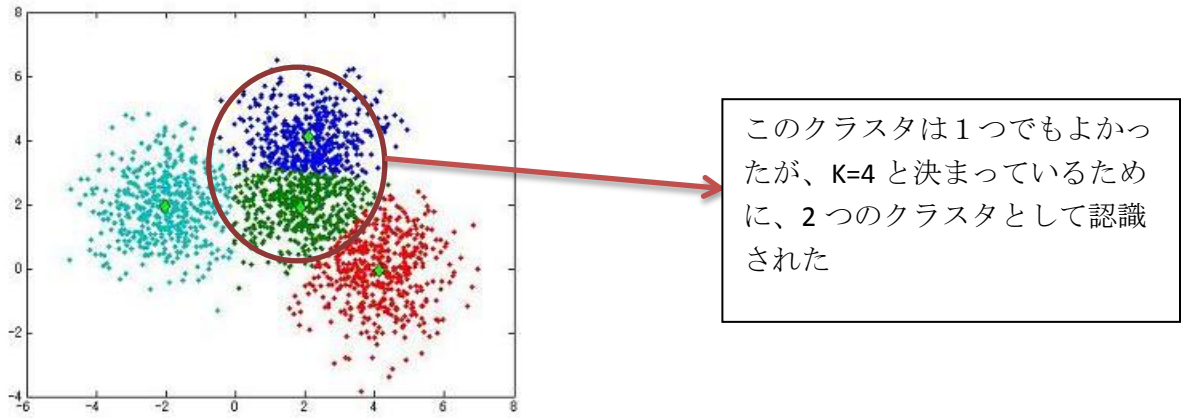

図 2.3.2　K-means が実行されたケース、K=4

Hierarchical Clustering

階層的クラスタリングでは、結果は樹状図（デンドログラム）として表される。
手順を次に示す。
1. あらゆるクラスター、対象間の距離を求め、最も近いものを新しいクラスターとする。
2. 新しく形成されたクラスターとその他との距離を求める。全てのクラスター、対象間の距離のうち最も近い２つを結合して新しくクラスターを作る。
3. 全てのクラスター、対象が一つのクラスターに結合されるまで繰り返す。

なお、この際にクラスタ間の距離の定義が必要になるが、これには典型的には以下の３つの距離がよく用いられている。

１．最短距離法

これは２つのクラスタのメンバー間で最も短い距離をクラスタ間の距離として定義する

２．最長距離法

これは２つのクラスタのメンバー間で最も長い距離をクラスタ間の距離として定義する

３．群平均法

これは２つのクラスタのメンバー間での平均距離をクラスタ間の距離として定義する

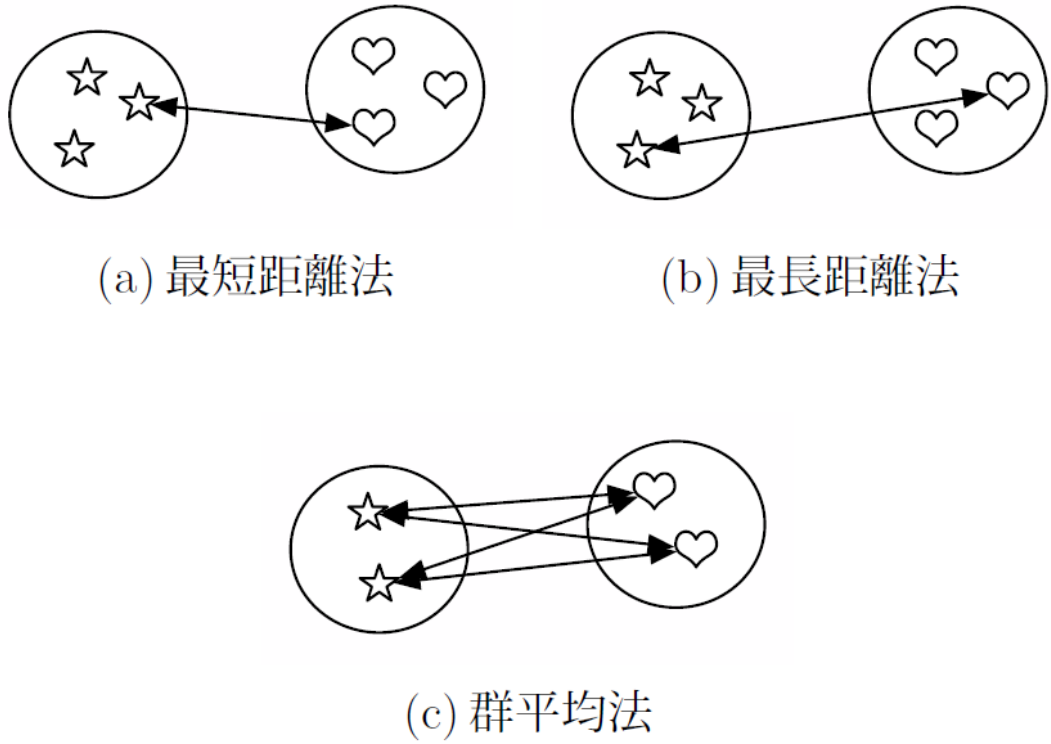

図 2.3.3　階層的クラスタリング手法のクラスタ間距離定義

この手法では、クラスタを階層的に形成するのだが、最終的なアウトプットとして本研究で必要なのは同じレベルのクラスタなので、これには様々な方法で得られる。

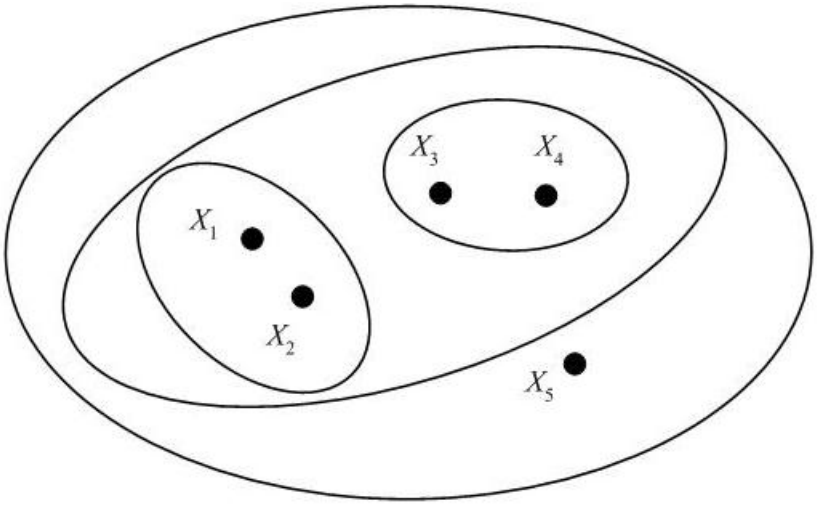

図 2.3.4　階層的クラスタリング手法の例

2.4　インターネット新語の性質とその分類

近年は世界各国でインターネット新語や隠語が生成され、コミュニケーションや情報交換に困難をきたしており、研究対象となっている。[13]

インターネット新語は文法制限を受けない、口語的な表現などの特徴があり、中国では爆発的なスピードで普及している。筆者はインターネット歴１０数年を持っているが、いまだに新語に迷うこともしばしばである。

中国語のインターネット新語はその機能の違いで専門家によって３種類に分けている。

１．インターネットそのものに関わる専門用語、たとえば BBS、在线（オンライン）などである

２．インターネットの出現によって現れた新語、例としては网虫（英語で言う netizen）や网瘾（ネット中毒）などである。

３．インターネットとは必ずしも関係性がなく、一般的な意味を持つ新語、例としては神马（What,Why などの意味）、给力（おおよそ「良い」といった意味）などの単語がある。特にこの３番目のインターネット新語はインターネットにとどまらず、日常生活にも広く浸透しており、新聞雑誌にも載っているケースが目立っている。

私の研究では特にこの３種類のインターネット新語を分けずに、すべて網羅的に研究の対象とするが、特に興味深いのはやはり３番目である。


人民日報
RENMIN RIBAO
人民网
网址:http://www.people.com.cn
手机:http://wap.people.com.cn
2010年11月
10
星期三
人民日报社出版
CN 11-0065
今日24版
改革攻坚进发动力 政策创新激发活力 厚积薄发释放能力
江苏给力“文化强省”

《人民日报》昨天头版版样截屏

図 2.3.1 中国共産党の機関紙に載っているインターネット新語「给力」

研究をする前に、中国語のインターネット新語にはどういった特徴があるのかをひとまず以下のように把握しておく。

中国語の新語の特徴を二つの側面で考察してみよう。１．新語の言語的な特徴　２．新語の生成パターン

１．新語の言語的な特徴

インターネット新語を言語的な特徴に基づいて分類すると、以下のように分類される。

① 既存の単語に付与される新たな意味
このジャンルの単語は例として「冲浪」があり、これはサーフィンの意味だが、日本語と同じようにネットサーフィンの意味として使われている。さらに面白い例としては「蛋白质」があり、これは「笨蛋」「白痴」「神经质」の３つの単語が略として合わせて作られた。意味としては「バカ」「アホ」「神経質」が足したものになる。

② 漢字による新語
a. 音声の変化による単語、例としては「神马(shen ma)」（what,why などの意味だが、もともとの単語は「神々しい馬」）が「什么(shen me)」から変化してなった。
b. 同じ発音の別の当て字による単語、例としては「斑竹(ban zhu)」が「版主(ban zhu)」という意味になる。

③ インターネット符号
a. 英語の略語、そのほとんどが漢字単語のローマ字の頭文字をとったもの。例としては「BT」が「变态(bian tai)」（変態の意味）になるなど。一部は英単語の略語で、英語圏でも使われている。たとえば「BF」は「Boy Friend」の意味など
b. 絵文字。こちらは日本語とほぼ通じるもので、感情や心境を表す。:-）など。これらは厳密には単語ではないので、今回も対象外とする
c. 数字あるいは数字と英文字を合わせたもの。こちらは中国語（ごく一部に英語など）の発音を数字などに当てた新語である。たとえば 88 や 3q など日本でも用いられている。

２．新語の生成パターン

これまでの新語は前述の言語的なトリックであるケースが多かったのだが、近年になって社会現象を取り入れた新語が流行しはじめ、定着するようになっている。たとえば「打酱油」（直訳すると、醤油を買いにいく）という単語は「自分とは関係ない」という意味で使われている。理由としては、以下のような経緯があった

2008 年初頭、そのころ中華圏のメディアを騒がせていた「陳冠希わいせつ写真流出事件」（通称「艳照门」事件）に関して、とある市民が街頭で広州テレビ局の記者に取

材を求められたところ、「关我屌事，我出来买酱油的！」（俺と関係ねーんだよ！醤油を買いに来ただけだ！）とカメラに向かって発言し、話題になった。

このように、多くの単語は単なる言葉遊びの域を超えて、社会現象と結びつくことで人々の支持を得ている。今ではインターネットのそれぞれのコミュニティーによって毎日のように新語、あるいは隠語が作られている。

図 2.3.2 「打酱油」の出所であるテレビ局のインタービュー

第 3 章

研究用データの取得と前処理

3.1　クローリング

3.1.1　ターゲットサイト

今回のクローリングに際して中国でもっとも活発なウェブ掲示板である MOP,TianYa,SMTH の三つのサイトを選んだ。それぞれの参考 URL と紹介は以下に述べる。

MOP:

中国でもっとも人気のあるウェブ掲示板の１つ、ユーザアカウント数５０００万個、一日のサイトアクセス数は２億ページ以上に達する。最初は１９９７年にゲームプレイヤーのための情報交換掲示板として立ち上がり、そのあとも勢いを持ったまま成長し、現在では中国屈指のインターネット新語、文化の発生地となる。

URL: *www.**mop**.com/*

図 3.1.1　MOP のサイト

TianYa:

１９９９年中国海南省で誕生した総合ウェブ掲示板、現在ではユーザアカウントが５６００万個に達し、常時オンライン人数が１００万人を誇る中国トップのウェブ掲示板。グーグルとも一時期パートナーシップを組んでいた。こちらのウェブサイトはコンセプトとして中国大陸のみならず全世界に居る華人をも対象としているどころが特徴である。

URL: *www.tianya.cn/*

図 3.1.2　TianYa のサイト）

SMTH:

中国のトップ大学である清華大学の学生たちが立ち上げたサイトが元となる中国知識人の間ではもっとも有名な掲示板である。アクセス数などは MOP や TianYA にはかなわないが、情報量やオリジナリティーはかなり高く、質の高い掲示板の代表格である

URL: *www.newsmth.net*

図 3.1.3　SMTH のサイト

3.1.2　ウェブクローラ

今回は上述の３つのサイト掲示板からランダムにページをクロールし、実験に使う。使うウェブクローラはスクリプトではなく、市販の製品 **Offline Explorer Pro** を採用している。

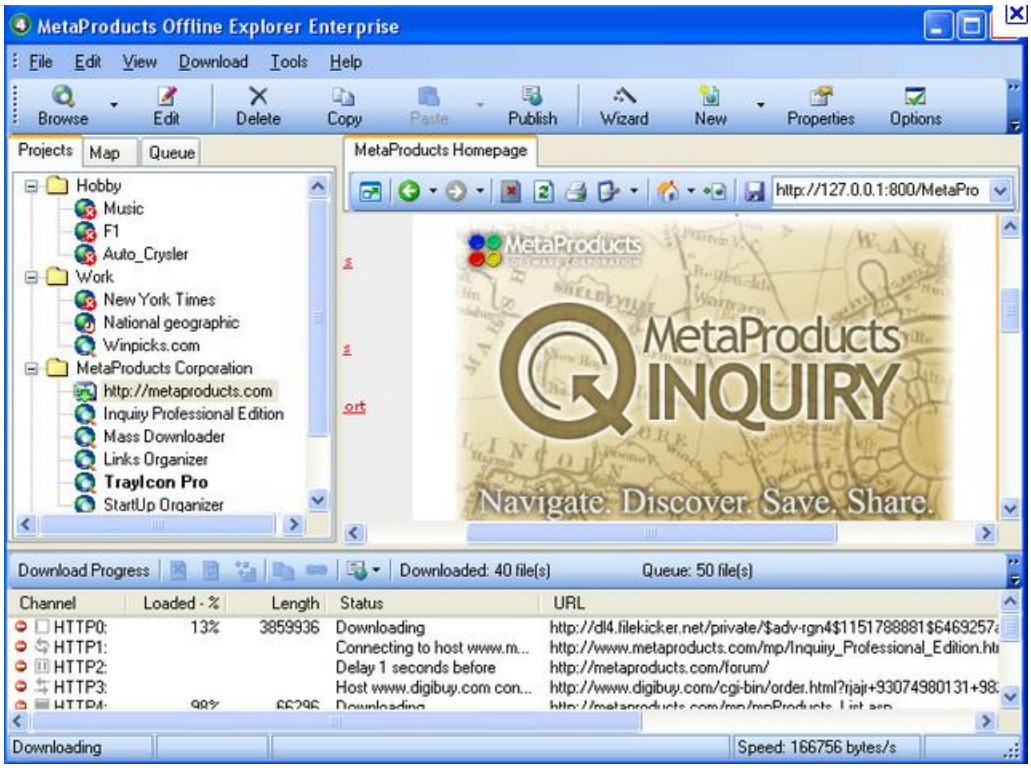

図 3.1.4 **Offline Explorer Pro**

これらのサイトからクローリングしてきた HTML などのテキストデータ総量は 2.5G で 6.5 万個のファイルになった。これまでの LSA 研究で使っているファイル数からするとこれくらいのサイズは最大級となるので、このようなデータで十分だと考えられる。

実際の LSA では行列の SVD を行うステップがあり、計算時間が長いため、10 万個以上のファイル数は今までの論文でもなかった。しかし 6 万個くらいのファイルを対象にした研究では有意義な結果が得られたので、本研究でもこれくらいのデータ量が十分だと考えれれる。

3.2　ワードプロセッシング（ICTCLAS）

クローリングしてきた HTML データは以下のステップで LSA にかけられる形にしていく。

a. HTML タグを取り、テキストファイルにする
b. ICTCLAS によって形態素解析を行い、単語の集合ファイルにする
c. ストップワードなどを除去、処理に適した形にする

これらのタスクは、a と c は Python で行なっており、b は ICTCLAS のライブラリを使うために Visual C++でプログラミングしてある。以下にその概ねのアルゴリズムを示す。

Input_f:Html ファイルのストリーム
onechar:ストリームにある 1 つの文字

```
for onechar in input_f:
    if onechar=="<":
        flagtag=False
#        print "'<' found,flag set to false\n"
        continue
    if not flagtag:
        if onechar==">":
            flagtag=True
#            print "'>' found,flag set to true\n"
            continue
    if onechar=="&":
        flagsign=False
        signcount=signcount+1
        continue
    if not flagsign:
        if onechar==";" or signcount>=maxlen:
            flagsign=True
            signcount=0
            continue
        else:
            signcount=signcount+1
    if flagtag and flagsign:
#        print "flag is true,ready to print %s\n" % onechar
        outfile.write(onechar)
```

図 3.2.1　HTML タグ処理プログラム

```
        //  printf("the output file name is %s\n input is %s\n",output,input);
            ICTCLAS_FileProcess(input,output,CODE_TYPE_GB,false);
        }
        }
    } while(FindNextFile(hFind, &FindFileData));
    FindClose(hFind);
  }
  else{
    cout << "ファイルがありません(" << sFindDir << ")\n";
  }
  return iFiles;
}

int main(int argc, char* argv[])
{
// char* sResult;
// bool fileprocessed;
    if(!ICTCLAS_Init())
    {
        printf("Init fails\n");
        return -1;
    }
    else
    {
        printf("ok\n");
    }

/*  int nPaLen=strlen(sParagraph);
```

ICTCLAS_FileProcess 関数は

ICTCLAS の形態素解析機能を

ラップアップしたインターフェース

図 3.2.2　中国語形態素解析プログラム（C++）

さらに、処理が完了したファイルから、サイズが 4KB 以下のものを削除し、最終的にサイズにして 600M の 3.1 万個のファイルが得られた。

```
t=_right><font color='#228b22'>整风严打公告</font></a><br />
l" target=_right><font color='red'>人肉搜索历年重大事件(2001-2008)</font></a><br />
l" target=_right>解密"人肉搜索"</a><br />
l" target=_right><font color='#0000FF'>人肉区发帖须知</font></a><br />
l" target=_right>善用网络利剑---人肉搜索！ </a><br />
l" target=_right><font color='#FF1493'>赏金猎人六大特色主题群大招募</font></a><br />
t=_right><font color='red'>人肉搜索区爱心扫盲(新手教学)</font></a><br />
t=_right><font color='#8b008b'>版主说在前面的话</font></a><br />
```

図 3.2.3　処理前の生データ例（Html ファイル）

楼 主 刘翔真 老 伤 弯道 技术 以前 差 一点点 … 回复 球 感 差 很多 哎 罚

猴子 _ MM 审核 看 母 猪 有 双眼皮 审核

図 3.2.4　処理が完了したファイルの一例、赤く囲まれているのは新語

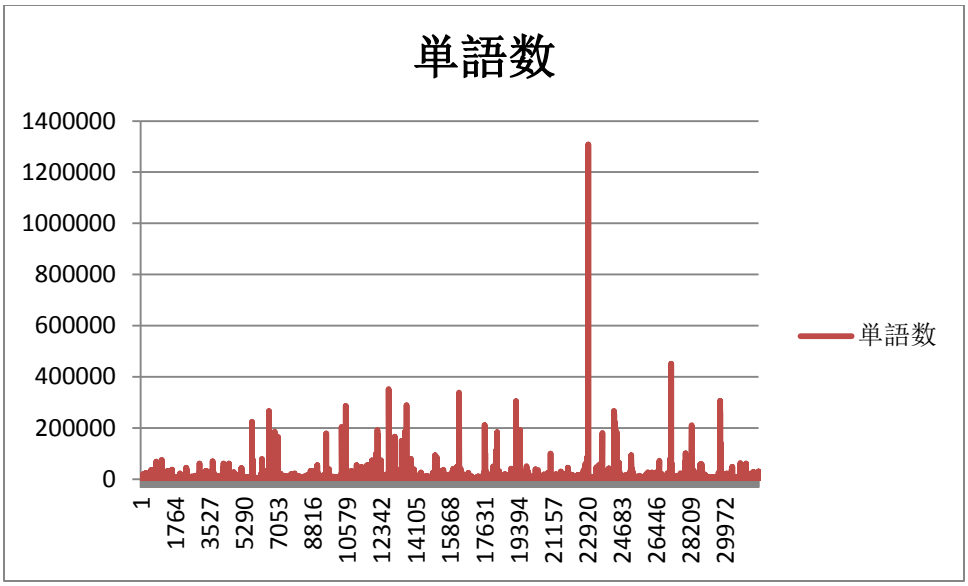

図 3.2.5　処理が完了したファイルにある単語数の分布

図 3.2.5 でわかるように、ファイルに含まれる単語数の分布が不均衡である。単語ごとの平均単語数は 4188.656 個である。なお、0 個や 1 個単語しか含まないファイルも存在している。このようなファイルごとの単語数のばらつきはクローリングの随意性や、掲示板の Html ファイルごとにトピックスが複数存在することなどに起因している。

第 4 章

潜在的意味解析による新語分析

4.1　LSA による新語分析の概要

LSA は潜在的意味解析と呼ばれ、単語の co-occurrence に依存しないのが大きな特徴である。これは正しく新語の意味解析に適しているのではないかと考えられる。何故ならば、新語は既存の単語の代替として使われる場合、既存の単語と一緒に現れるというよりは既存の単語と似たコンテキストに現れることが想像できる。

本研究ではインターネット新語を予めリストアップし、クローリングしてきた文書にある７８個の新語を研究対象としている。

```
2060: 擦  :4960
2448: 火星  :4034
2521: BT  :3891
2602: 沙发  :3771
2714: 浮云  :3603
2878: 粉  :3327
3248: pp  :2883
4028: 囧  :2168
4688: YY  :1774
4738: 斑竹  :1747
4862: 潜水  :1689
5326: 包子  :1502
5566: PP  :1429
5641: sigh  :1398
5847: 马甲  :1336
6321: 粉丝  :1206
6549: BH  :1152
6888: lg  :1073
7054: LG  :1043
7191: 天然  :1017
8090: 大虾  :872
8650: PS  :795
```

図 4.1.1　インターネット新語のリスト

研究方法としては、頻度が１００を超える単語を対象にし、LSA でウェブテキストデータを単語特徴ベクトルを用いて、以下の２つの手法で新語の特性を調べてみた。

１．新語とコサイン距離が近い語の抽出

この手法は数々の研究で synonym あるいは similar words を見つけるのに用いられている。[1][2][3][4][10]

２．単語空間でのクラスタリング分析

以下の節ではそれぞれの理論根拠及び実験結果について述べる。実験全体の流れを以下に図示する。

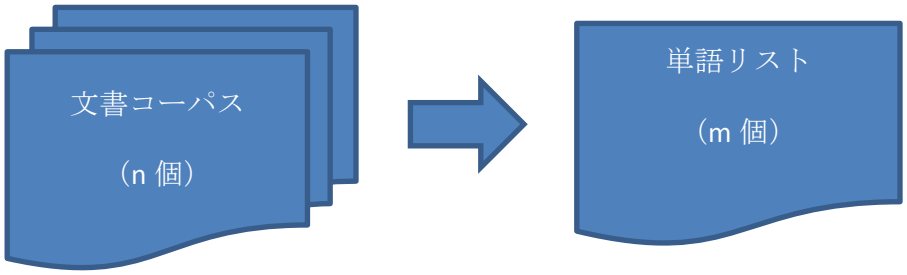

図 4.1.2　ステップ１　文書コーパスから単語を抽出

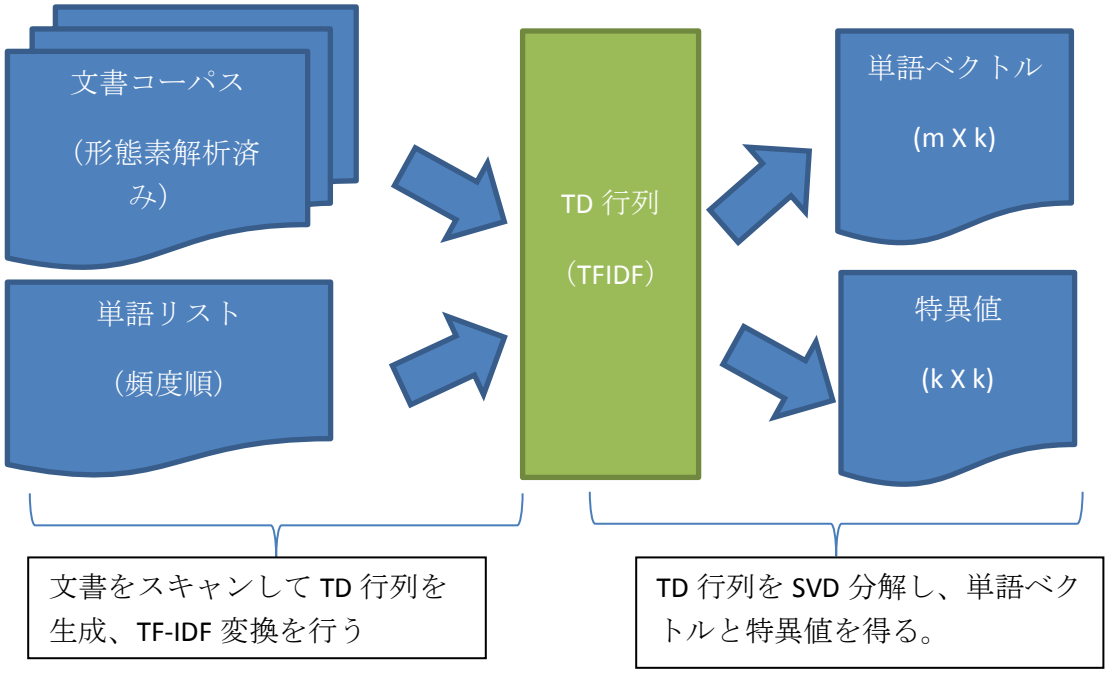

図 4.1.3　ステップ２　文書から単語ベクトルを生成

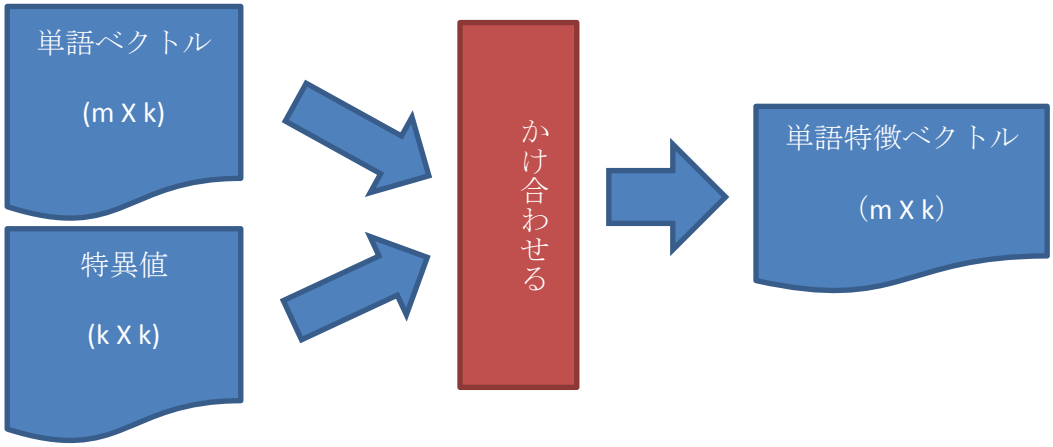

図 4.1.4　ステップ３　単語特徴ベクトルの生成

4.2　SVD による単語ベクトルの取得

本研究では SVD を行うために、python の scipy,numpy と alglib の３つの科学技術計算ライブラリを用いている。Scipy にも SVD の関数があるが、度々の実験によって、大きな行列の計算ではバグが発生するので、代用として alglib を使っている。Scipy に含まれる行列計算関数などは使用している。

SciPy

SciPy は Python のための科学的ツールのオープンソース・ライブラリとして開発されている。SciPy は配列の高速な操作のためのすべてのライブラリを含んでおり、人気の Numeric モジュールを置き換え、ひとつのパッケージとして高レベルな科学と工学のモジュールを集めたもの。

SciPy は、配列オブジェクトとその他の基本的な機能を備えた NumPy を基礎にしている。SciPy は統計、最適化、積分、線形代数、フーリエ変換、信号・イメージ処理、遺伝的アルゴリズム、ODE (常微分方程式) ソルバ、特殊関数、その他のモジュールを提供する。

NumPy

NumPy は Python プログラミング言語の拡張モジュールであり、大規模な多次元配列や行列のサポート、これらを操作するための大規模な高水準の数学関数ライブラリを提供する。初期のバージョンは Jum Hugunin によって作成されたが、NumPy はオープンソースであり多数の開発者が寄与している。

Python はインタプリタ言語であり、数学のアルゴリズムは C 言語などのコンパイル言語や Java な どと比べて低速に動作する場合が多い。NumPy はこうした問題を、多次元配列と、配列を操作する多数の関数や演算子を提供することでこの問題を解こうとしている。これにより、配列や行列の操作として記 述できるアルゴリズムは、等価な C のコードとほぼ同等の速度で動作する。

MATLAB のプログラミング言語は表面上は NumPy に似ているため、NumPy をフリーの MATLAB の代替物として使用するものもいる。いずれもインタプリタ言語であり、いずれもスカラーで はなく配列や行列に対する操作を高速に行うプログラムを書くことができる。それぞれの特徴としては、MATLAB は高価だが、組み込みの数学関数を多数備え、さらに様々な用途のための実用的なパッケージが提供されているという利点がある。一方 NumPy は、ベースとしている Python がそもそも現代的で完全な汎用のプログラミング言語であり、習得が容易で生産性が高いことが最大の利点である。加えてオープンソースでフリーである。SciPy は NumPy にさらに MATLAB 的な機能を追加するライブラリであり、Matplotlib は MATLAB に近いグラフ機能を提供するパッケージである。

Alglib

Alglib はマルチプラットフォームに対応した線形代数のライブラリであり、C/C++や python,JAVA など様々な言語をサポートしている。これは ALGLIB Project という会社で開発されているものだが、研究用にはオープンソース契約で商用には商用契約となっている。このため、商用にも耐えうる品質を持つ。
本研究では SciPy の SVD 関数が大きいデータに対応できないため、Alglib の SVD 関数を用いる。

本研究では C++による実装も試みたが、Python による実装とのスピード差が出ないため、形態素解析以外は全て python によるプログラミングに終始している。

```
116
117  print "starting to do the svd:"
118 # TD_U, TD_S, TD_Vt = svd(TDmatrix)
119  tempmatrix=TDmatrix.tolist()
120  tempm,tempn=TDmatrix.shape
121  del TDmatrix
122  done,S,U,Vt=rmatrixsvd(tempmatrix,tempm,tempn,1,0,2)
123  del rmatrixsvd
124  if done:
125    print "svd done successfully"
126  else:
127    print "svd failed,exiting"
128    exit(1)
129
130  del Vt
131  TD_U=asarray(U)
132  del U
133  TD_S=asarray(S)
134  del S
135  #TD_Vt=asarray(Vt)
136
```

rmatrixsvd 関数が SVD 分解のインターフェース。
実際使われる行列は Numpy の array クラス

図 4.2.1　alglib の SVD 関数によるコーディング

特異値が合計で全体の 50%以上を占めるように次元削減を行なっており、結果としては 31000 次元だったベクトルが 2459 次元で済んだ。最大の特異値は 10.6 で最小の特異値は 0.196 である。2459 個の特異値を全て足し合わせると 1131 に達するが、トップ 500 個を足すと 562 を超え、約全体の半分を占めている。

4.3　意味が近い語の抽出

4.3.1　Similarity の計算とその比較

高次元ベクトル間の関係性の強弱を示す指標として、Similarity が定義されている。数学的には、ベクトル間の角度を示す Cosine Similarity がよく用いられる。その他には、高次元ベクトルを高次元空間の点とみなして、ユークリッド距離や、マンハッタン距離などの距離の定義ができる。Similarity は本研究では概ね距離とは反対の概念に当たる。

それぞれの Similarity(距離)の定義とその特徴を紹介する。高次元空間にベクトル $U(x_1,x_2,\ldots,x_n)$ と $V(y_1,y_2,\ldots y_n)$が存在する際に、それぞれの距離の定義は以下になる

ミンコフスキー距離

幾つかの距離がミンコフスキー距離の特殊形として与えられるので、まとめると以下のようになる。

マンハッタン距離
$$= \sum_{i=1}^{n} |x_i - y_i|$$

ユークリッド距離
$$= \left(\sum_{i=1}^{n} |x_i - y_i|^2 \right)^{1/2}$$

p-ノーム 距離
$$= \left(\sum_{i=1}^{n} |x_i - y_i|^p \right)^{1/p}$$

これらの距離は空間上の点の距離と見なすことができ、直感的な距離である。しかし、単語特徴ベクトルの長さは頻度に強く関係しており、本研究では単語の特徴ベクトルは長さよりも向きが重要で近似度を表すと考えられるので、以下のコサイン距離を用いることにしている。

コサイン距離

コサイン距離は以下の式で与えられている。

$$1 - \frac{uv^T}{|u|_2 |v|_2}$$

Similarity を計算する際はコサイン距離を 1 で引くことで求まる。

4.3.2　実験結果

実験では次元数と Similarity の閾値を変えながら、78 個の新語と関係が強い単語を抽出してみた。以下にこのリストを示す。１つ目の番号は新語が全単語における頻度順のランキングで、２つ目の数字は出現頻度を表す。

593: js :17790

631: LZ :16769

1105: 表 :9513

1206: 偶 :8925

1370: mm :7953

1420: QQ :7619

1698: 亲 :6200

1853: MM :5680

1974: NB :5229

2060: 擦 :4960

2448: 火星 :4034

2521: BT :3891

2602: 沙发 :3771

2714: 浮云 :3603

2878: 粉 :3327

3248: pp :2883

4028: 囧 :2168

4688: YY :1774

4738: 斑竹 :1747

4862: 潜水 :1689

5326: 包子 :1502

5566: PP :1429

5641: sigh :1398

5847: 马甲 :1336

6321: 粉丝 :1206

6549: BH :1152

6888: lg :1073

7054: LG :1043

7191: 天然 :1017

8090: 大虾 :872

8650: PS :795

8855: dd :770

10116: 水母 :637

10496: 稀饭 :604

10550: 鸭梨 :599

10627: 蛋白质 :594

10831: lp :579

11300: rp :547

11368: TMD :543

11532: 虾米 :532

12158: 片片 :492

12759: PK :461

12947: cool :452

13024: XXX :448

13095: DD :445

13252: KFC :437

14357: BB :392

14508: nb :386

14996: 水仙 :370

15542: LP :352

16172: MD :334

16199: 拜拜 :334

16259: 米粒 :332

16680: BS :321

17246: 嘲讽 :307

17833: btw :294

19576: bs :260

19942: ML :253

20757: BF :240

22415: 酱紫 :216

23435: RP :205

24464: SP :194

25032: JS :188

26622: pm :172

28359: 潜水员 :157

28845: SM :154

28974: 水桶 :153

29220: GF :151

29263: BL :151

30297: 果酱 :143

31453: XXOO :136

31455: kao :136

32117: bl :132

32949: 筒子 :127

33371: DL :125

33910: 小强 :122

34147: tg :121

36850: 白骨精 :107

以上のリストはウェブにある新語リストを元に、今回の対象となる 3.8W 個の単語に含まれるものを抽出したものとなる。ウェブにある新語リストは 200 個以上あったので、今回がカバーできているのは約 1/3 にあたる。このリストには様々な種類の新語が含まれており、分類すると以下のようになる。

1.英文の略語
2.複数の漢字からなる単語
3.漢字一文字

後で述べるが、これらの新語の性質はかなり異なり、結果に大きな影響を及ぼしている。

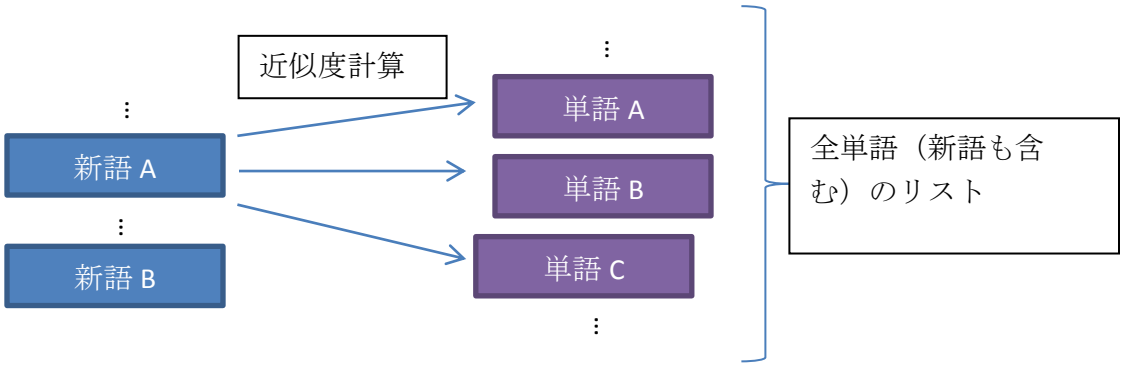

図 4.3.1　実験の仕組み：全ての新語対して、全単語との近似度を計算

新語との近似度が閾値以上の単語を新語の意味群として抽出する。本研究では閾値の近似度を 0.7 として結果表示しているが、実際の応用は閾値を変えることで結果をある程度広い範囲で見ることができる。

実験の結果は全体として、英単語は英単語と高い近似度を示しており、その多くは Javascript などのメタデータであるため、ノイズが多い。新語の近似語としても多くの英単語ノイズが入っている。さらに、LSA は「意味」ではなく、とある単語に含まれるコンセプトを測るので、新語に近い単語であっても、必ずしも意味が一緒ではない。例えば「ソファー」に意味が近いのは「椅子」だが、「室内」などもかなりの近似度が見られる。新語は新語と近似度が高いことも見られ、それはすなわち新語をよく使う人の存在を示しているものだと考えられる。新語をたくさん使う人と、新語を全く使わない人が居るようだ。

さらに、次元が増えるに連れて、近似度がかなり異なるものもあるが、近似度が上位であるものは Similarity の数値が変わっているが、上位にあることが変わらないケースが全体的に多かった。しかし、次元があるにつれて、近似度が全体的に下がっており、抽出される単語も減る傾向がある。これはコンセプトや次元数が増加することによってもたらされる自然な結果であり、LSA の他の論文でも同じ現象がわかっている。

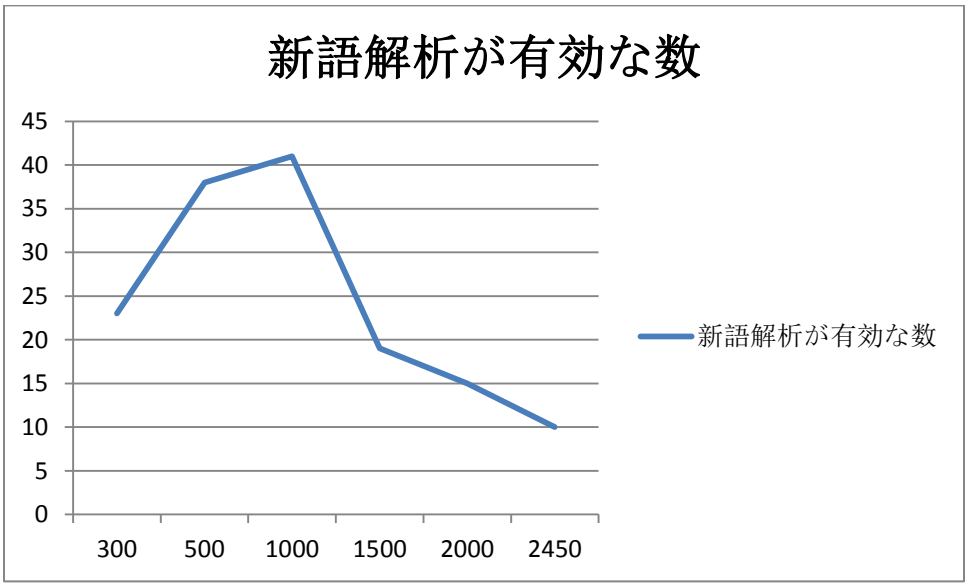

図 4.3.2　強く関連する単語が見つかったケース

図 4.3.2 では、主観的な判断ですが、新語 1 つに対して近似度の上位 10 語に 5 語以上の有効的な意味群語が見つかった結果を集計している。

以下は実験において特に興味深い結果の一部を示して説明していく。

PK:
赛区: 0.71683970885

対戦グループ

図 4.3.3　強く関連する単語が見つかったケース

PK は Player Killing の略で、オンラインゲームの用語として普及し、今は「対戦」や「喧嘩」の意味を持つ。それに対しての単語は「賽区」、つまり「試合の場所」というオンラインゲームでよく見かける用語である。これは見事に関連性のある単語を見つけた。

```
RP:
罩: 0.743283965273
人品: 0.717490233452
British: 0.703884963065
```

覆うなどの意味を持つ漢字
品質

図 4.3.4　元々の単語が見つかった結果

RP は「人品（Ren Pin）」の略で、見事に元々の単語が見つかったのである。この単語は道徳から転じて、「運」の意味になる

```
BF:
纠缠: 0.746558537772
吼: 0.743919571756
panricoo: 0.713900962301
```

付きまとう
怒鳴る

図 4.3.5　人間の性質が反映された結果

BF は欧米でも浸透している略語で、Boy Friend、すなわち「彼氏」という意味だが、これに関連する単語としては「付きまとう」「怒鳴る」などであった。

```
火星:
探测器: 0.895566190492
子: 0.830791670674
围剿: 0.829579350432
我: 0.819738841631
女: 0.779788427079
宇航局: 0.731617700602
吓: 0.728149323611
表达方式: 0.706053887422
```

火星
探測器（宇宙）
「子」という漢字
囲い込む
「我」という漢字
「女」という漢字
宇宙局（日本でいう JAXA）
「不」という漢字
「表現方法」

図 4.3.6　本来の意味と新語としての意味両方見つかったケース

「火星」というのは日本語と同じ惑星の「火星」以外に、「火星文」という単語で使われたりする際に、「意味不明な言葉」を意味する。この例では関係性のある全ての単語がこの２つのグループにハマる。それぞれの例を違う色で囲んで、示している。

JS:
2009-10-24: 0.997150049123
Javascript: 0.99370730721
VSS: 0.984821687395
VBS: 0.974355101674
prototype: 0.965352653565
Ajax: 0.958717184896
SQLServer: 0.946892101251
悠: 0.938315864913
蓝天: 0.933327948542
WebServices: 0.933089509636
weblogic: 0.932298310321
09:31:15: 0.932215566232
jdbcODBC: 0.932130453327
箋: 0.931727910429
Question: 0.931193545129
database: 0.931071695137
DWR: 0.930633375924

「悠」
青空

「箋」

図 4.3.7　失敗例、新語の意味がある単語がほぼなし

JS は中国語で「奸商（Jian Shang）」の略で、「悪徳商人」の意味だが、ここでは Javascript の略である JS と混同されてしまい、英単語のノイズ（javascript の用語）がたくさんある以外、予想していた意味が見つかっていない。

沙发:
客厅: 0.786303005714
卧室: 0.780976774209
嘀咕: 0.757547646912
灯泡: 0.752379200372
床头: 0.748516059822
什么的: 0.747360779283
屋里: 0.745520447029
柜子: 0.73811144288
螺丝: 0.737970039767
秀气: 0.735356860574
晾: 0.731447709017
我家: 0.726068133027
墙上: 0.719673618367
开灯: 0.714993357686
侃: 0.709238373198
窗帘: 0.708975672088
样子: 0.700189882294

ソファー
ラウンジ
寝室
文句を言う
ランプ
ベッド
「などの」
室内
タンス
ネジ
優雅
干す
我が家
壁
ランプをつける
雑談
カーテン
様子

図 4.3.8　普通の意味しか見つかっていない例

「沙发」は「ソファー」という意味だが、新語としてはネット掲示板で一番目に返事をしたことを意味する。今回の実験では「室内」や「カーテン」など、元々の意味と近い単語しか見つかっていない。

また、本研究では漢字一文字からなる新語は有意義な結果をほぼ得られなかった。理由としては漢字一文字は中国語の性質上様々な場所に現れることや、ICTCLAS などの形態素解析プログラムは漢字一文字を正しく切り出せていない上に、漢字はとりわけ多義的で、様々な全く関係しないコンテキストに現れるからだと考えられる。

4.3.3　結果の評価

今回の実験では、次元数が 500-1000 の間で一番いい結果が得られた。与えられた 78 個の新語について、半数弱は意味のある結果が得られた。次元数をより正確にすることで、さらに精度を上げることも可能だと思われる。今回用意した 78 個の新語はあらゆるコミュニティーによって作られ、人によって使う頻度がかなりばらつきがあるもので、実験対象のサイトではポピュラーじゃない新語ももちろんたくさんある。これらの現象も結果によって確認できた。

さらに、彼氏に関連するのが「怒鳴る」「付きまとう」などのマイナスの言葉なども個人的には興味深かったのである。英単語の新語は全体として結果がよくないのはデータの前処理で英単語のノイズをうまく除去していないことが原因だと考えられる。今後の実験ではメタデータの除去を徹底的に行うべきである。

総じて言えば、LSA による新語の意味検出はある程度成功しているが、データの性質に左右される欠点があり、まだまだ改善の余地がある。手法の有効性は十分に示せたものだと思われる。

4.4　クラスタリングによる新語分析

4.4.1　クラスタリング手法について

本研究では、単語特徴ベクトルをクラスタリング分析を行い、クラスタの分布から新語の生成コミュニティーを見つけることを目的としている。4.3 節の手法は新語に限定した手法となっているが、単語空間全体に対してクラスタリング分析を行うことで、新語と既存語の関係性を一層考察していく。

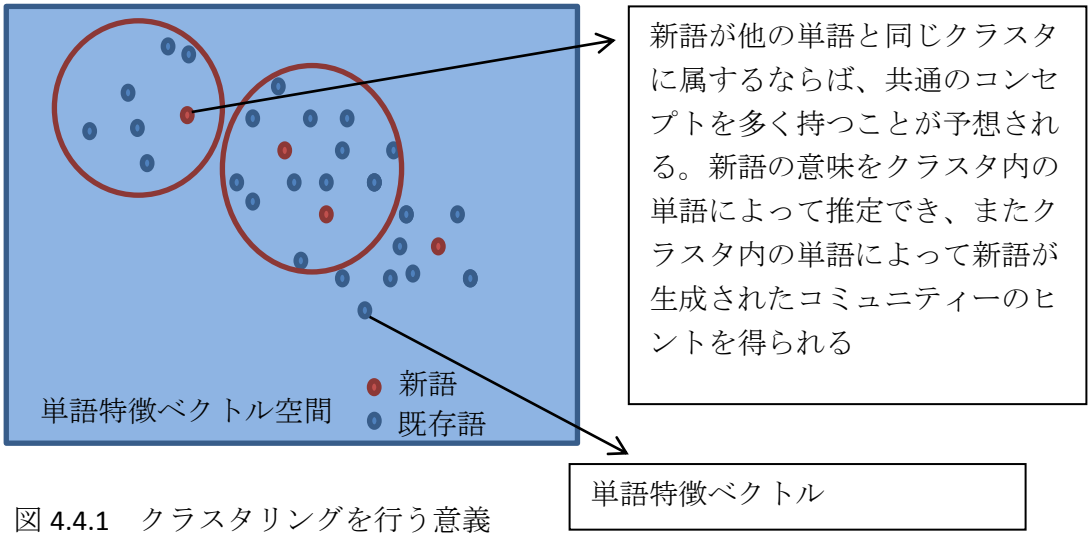

図 4.4.1　クラスタリングを行う意義

クラスタリングを行う上で、本研究ではパラメータ付きの手法である K-means とパラメータ無しの手法である階層的クラスタリング手法を使用している。K-means を使うと、予めクラスタ数を決めなければいけいないが、クラスタ数を示唆するような情報がないため、クラスタ数指定がない階層的クラスタリングと併用し、結果を解析、比較する。

なお、階層的クラスタリングでは距離の計算はユークリッド距離である必要がないため、4.3 節との整合性を持たせるようにコサイン距離を用いている。実験の結果は 4.4.2 節にて詳述する。

4.4.2　実験結果

K-means

K-means を用いた実験は 500 次元、1000 次元、1500 次元、2000 次元に削減した単語特徴ベクトルに対して行った。さらに、初期クラスタの指定を 78 個の新語にし、全部で 78 個のクラスタを形成するように実験し、ランダムに初期クラスタを選択、クラスタの数を変えながら実験を行った。

クラスタリングにおける次元の呪いを回避するように、100 次元や 50 次元などの削減した次元でも実験を行なってみた。

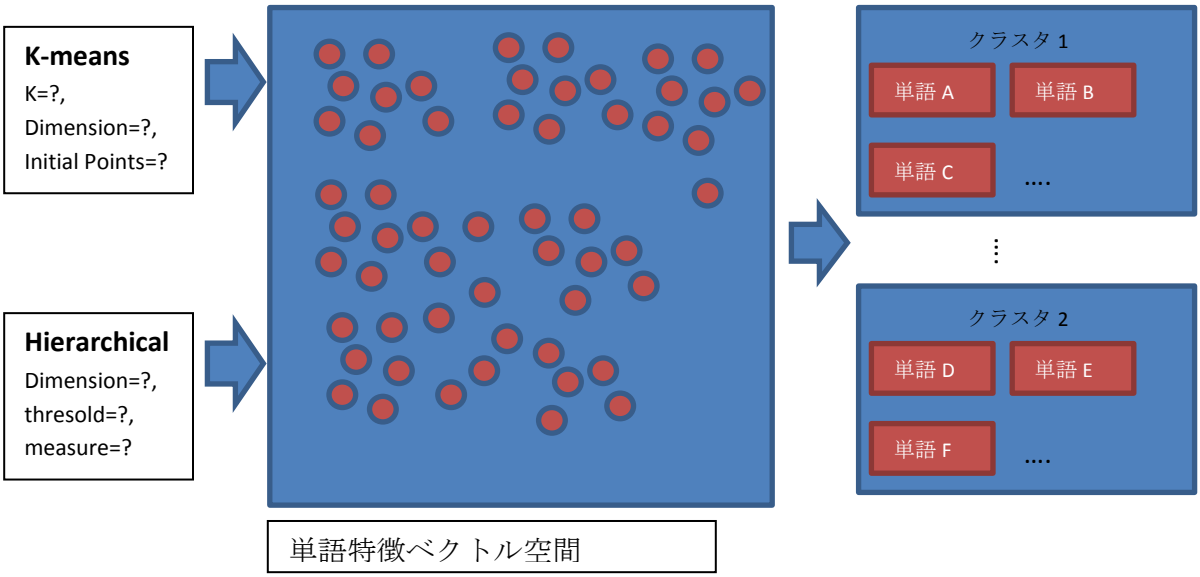

図 4.4.2　クラスタリング解析の仕組み

図 4.4.2 に示しているように、二通りのクラスタリング手法はそれぞれパラメータを決める必要がある。2 つの手法で共通するのは、単語特徴ベクトルの次元数を決めなければいけない。これは使う次元数によって結果が変わってくるので、パラメータをチューイングして最適な結果を得るために決めるべきである。

K-means 手法では目標とするクラスタ数 K と繰り返しがスタートする初期値を与えなければならず、階層的クラスタリングは距離の計算法とクラスタとして切る閾値を決める必要がある。

K-means のクラスタリング結果は一部のクラスタでは新語と密接に関係する単語が現れているが、全体としてクラスタ数が 500 から 2000 まで、次元数に関わらず、大きなクラスタ（２万から３万個の新語が含まれる）ができてしまい、ほとんどの単語が含まれている。この結果では予想に反し、半数以上の新語が同じクラスタに入っている。これでは結果が極限られたものになっている。

さらに、階層的クラスタリングではコサイン距離を用いているが、K-means の結果と同じように、極めて大きなクラスタが得られ、結果の解釈を困難にしている。

以下に K-means と階層的クラスタリング手法の結果を一緒に図に示す。

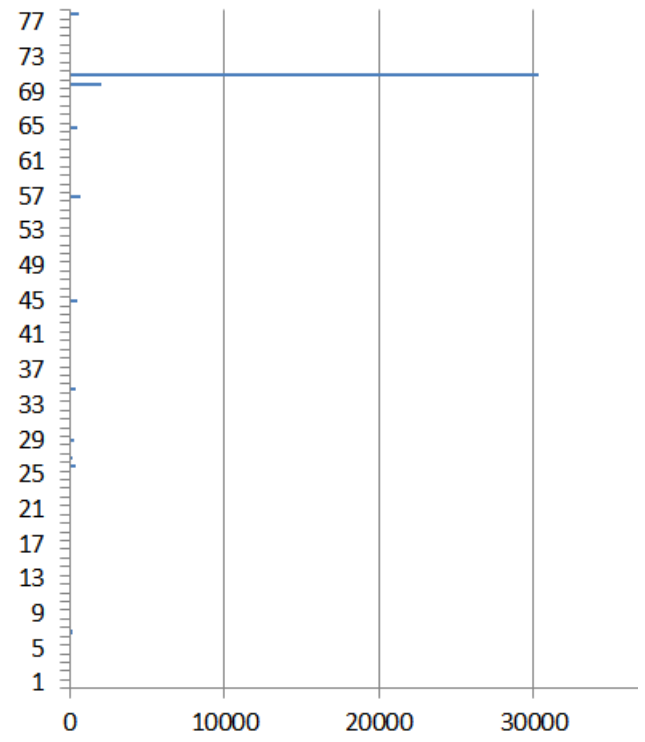

図 4.4.3　78 個のクラスタを指定するようでは大きいクラスタが形成される

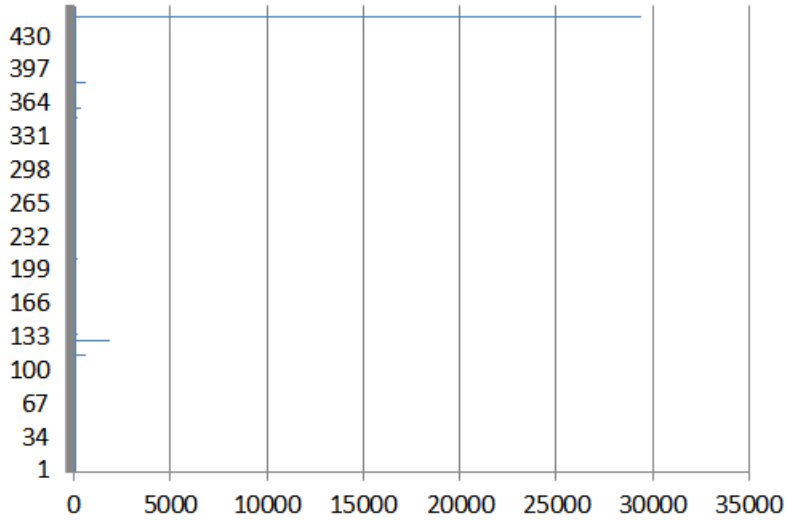

図 4.4.4　ランダムに初期クラスタを指定、1000 次元 500 個のクラスタを指定

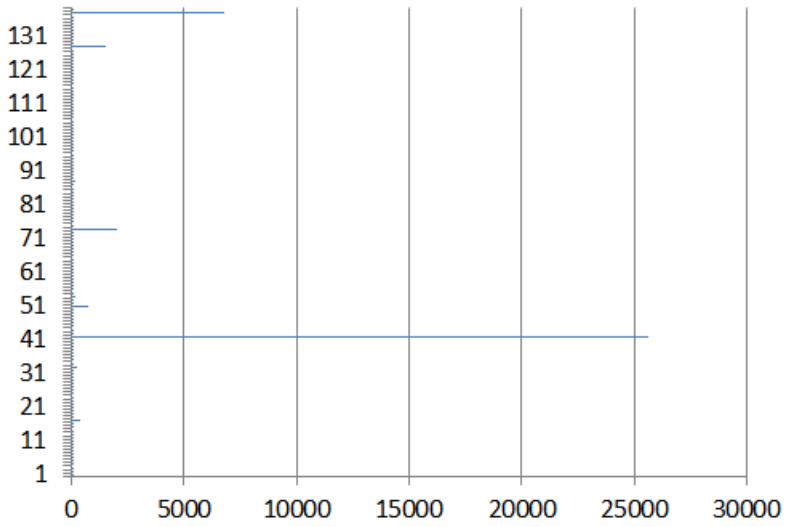

図 4.4.5　ランダムに初期クラスタを指定、50 次元、500 個のクラスタを指定

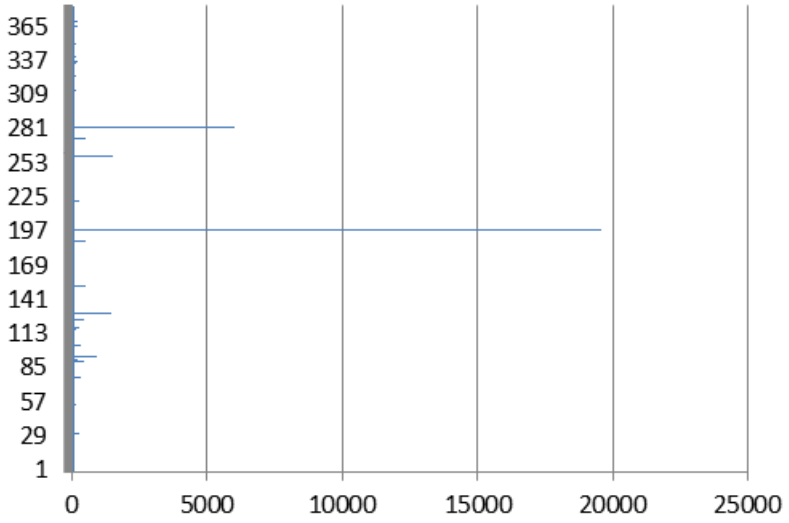

図 4.4.6　ランダムに初期クラスタを指定、100 次元、2000 個のクラスタを指定

大きなクラスタができてしまう一方で、単語数が適正のクラスタには意味のある結果が見られている。以下の図にそれを示す。

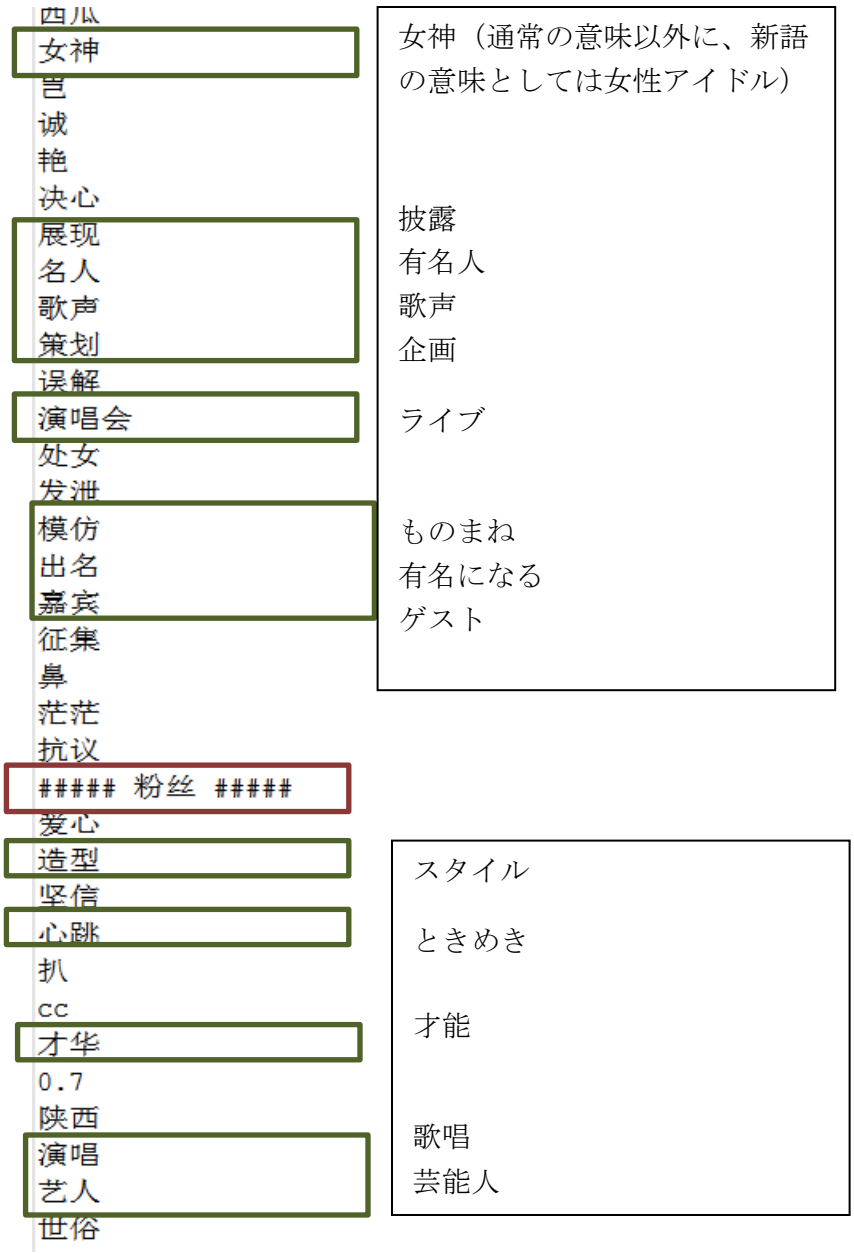

図 4.4.7　英語の「Fans」の中国語訛り「粉丝」が属するクラスタ

この図では中国語の新語である「粉丝」のコミュニティーや意味を的確に示している。この単語は辞書では「澱粉によって作られた透明な糸状の食べ物」の意味しかないが、このクラスタでは食べ物と関連する意味が全くない。一方で、4.3 節では単に「粉丝」に意味の近い単語の抽出では 0.7 以上の近似度を持つ単語が存在していない。

このように、大きなクラスタができることによって多くの新語の解析ができないものの、意味のあるクラスタも数多く存在する。これらの結果は 4.3 節の近似度による新語解析の補強になっていることが言える。

4.4.3　結果の評価

クラスタリングによる新語の意味解析では均衡的にクラスタが作られている一方で、パラメータや手法をいくら変えても、とてつもなく大きなクラスタが形成され、半数以上の単語が含まれている。一方で、単語が 1 つしかないクラスタがクラスタ総数の 1/3 以上を占めている。実際のクラスタ数を 10000 以上に設定してもクラスタ数が 500 以上にはならない。実際の単語は 3.8 万個あり、クラスタ数は人間でも決めることが難しいが、１つのクラスタに約 50-200 個の単語があると仮定し、クラスタ数が 2000-8000 ぐらいあるが妥当であろう。

このような結果になった理由としては、単語の頻度の偏りによるものだと考えられる。LSA では、単語の頻度が多いほど単語特徴ベクトルのノルムが大きくなる傾向がある。これは経験上単語の 1 番目の次元に現れていることがわかる。通常の距離計算がユークリッド距離ではなく、コサイン距離にした理由もこれにある。実験は単語特徴ベクトルの 1 番目の次元を取り除いて再度行う必要がありそうだ。

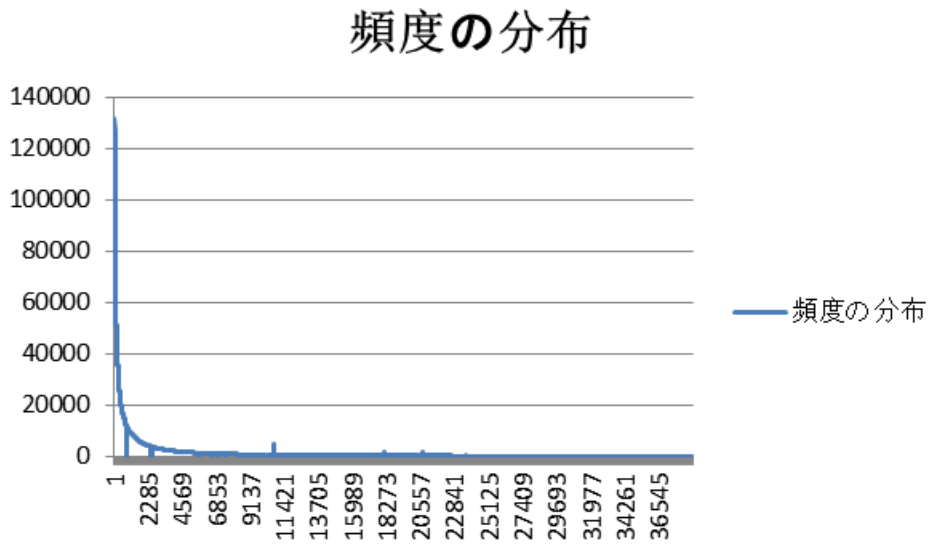

図 4.4.8　単語の頻度分布図

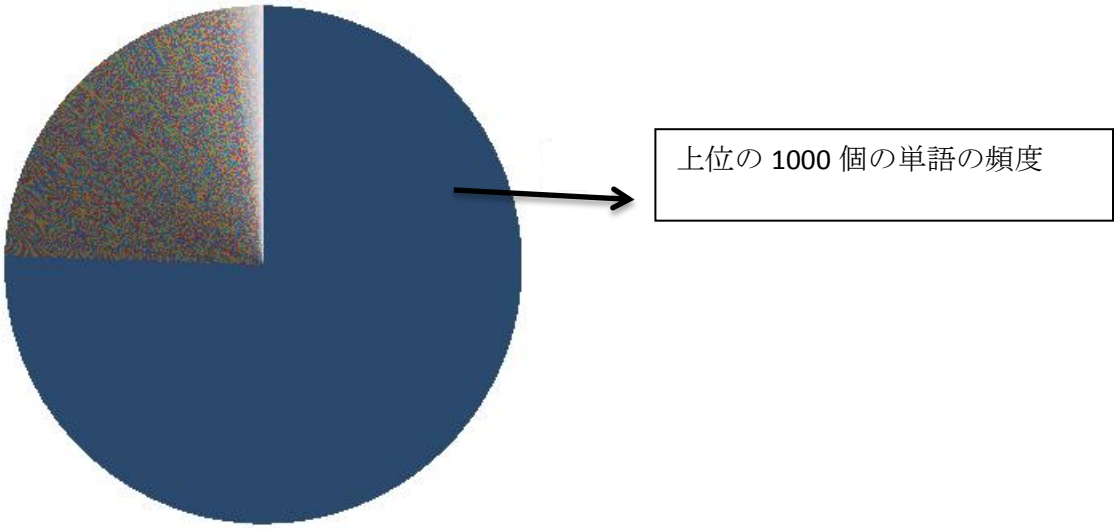

図 4.4.9　単語頻度構成図

これらの図からすると、上位の約 1000 個の単語が全体の頻度数の 3/4 以上を占めていることがわかる。ロングテールと呼ばれる単語の分布が見られる。このような分布では、単語ごとの情報量が圧倒的に異なり、クラスタリングがうまくいかないことの原因にもなるであろう。

今回クローリングしてきたデータには随意性があり、どうしても単語の偏りが出てくる。十分に大きなデータセットが与えられるとこのようなクローリングによる単語の偏りがいくらか解消されるが、やはり単語は頻出のものとあまり使われないものがある。

第 5 章

結論

5.1　実験結果からの示唆とまとめ

今回の研究ではインターネットから生のデータをクローリングすることから始めて、新語の解析まで一貫して行った。中国語のインターネット新語に対する LSA と ICTCLAS を用いた１つのシステムを確立したと言えよう。実験結果は新語の意味発見が実証され、LSA がインターネットの汚いデータや多義性を持つ新語にも対応できることがわかった。もちろん、実験の結果がまだまだ完璧とは程遠いが、データをより多く収集することで改善が可能である。

なお、LSA の弱点である膨大な計算量（$O(n^3)$）が度々指摘されるんだが、近年計算機の速度が格段に向上し、大型並列コンピュータを用いれば、$O(n^3)$はそこまで計算困難ではなくなりつつあるだろう。もちろん、LSA はインターネットの全データを解析したりするような処理には向かない。しかし、本研究のように新語の発生サイトを限定すれば解析するべきデータ量を抑えることもできる。新語の発見は中国語では形態素解析の難しさから難題とされるが、新語の発見技術を応用し、新語が存在するページのみあるいは特定の分野に限定することで解析するべきページ数を減らせるものである。分野が限定された新語を少しずつ解析し、最後に合わせれば全ての新語をカバーすることができる。

中国語には他の言語に見られない新語の普及が目立ち、いずれ人間による分析が追いつかなくなる日も来るだろう。人間の言語学者がインターネット新語の研究に本研究を応用すれば、研究のヒントになるだけでなく、１つの客観的な評価手法にもなると考えられる。本研究が提示した改善点を踏まえて、実用に耐えうる新語データ解析ツールが大いに役に立つと思われる。

5.2 今後の課題

本研究の不足と発展をいくつか指摘しておこう

1. HTML ファイルの処理は十分ではない

   本研究では HTML のタグのみを取り除いてある。XML ツリーによる意味解析を行なっておらず、メタデータが本文に混じっていることがある。これは実験結果では javascript のキーワードが極めて高い頻度で出てくることに起因している。

2. 単語頻度の極端な偏りには対処できていない

   頻度が極めて高い単語を除去するべきだが、数が少ないので、これだけで結果が飛躍的によくなることが想像しにくい。単語頻度の偏りを解消するには計画的に様々な分野からウェブページをクローリングしないといけいない。今後の研究はこの点に注意していただきたい。

3. 同じページに複数の概念が存在することもある

   今回クローリングの対象となるページはウェブ掲示板であるが、1 つのファイル（ページ）に複数の議論が存在するケースも見当たった。従来の LSA では 1 つのファイルではなく、1 つのパラグラフ（段落）を用いており、1 つのパラグラフだと同じ概念を述べていることがほとんどだが、ファイルの構成はサーバーサイドの都合によって複数の概念が存在している。この現象は本研究の精度に大きく影響を及ぼしているものと考えられる。

4. LSA 及びクラスタリングのパラメータを客観的に決める手法がない

   LSA では次元数の削減がコアな概念であるが、次元をどこまで削減したらよいかは実験によって決めるべきである。他の研究では Toefl の同義語問題を解かせて一番点数のよかった次元を選んだりしているが、これもやはり Toefl の問題次第で結果が変わるものである。つまり人間が介入することによる随意性が LSA では無くすことができない。クラスタリングは K や初期値、あるいは階層的クラスタリングでの閾値を決める必要があったが、これは元々のデータの性質をよく知って決める必要がある。今回は膨大なデータを処理しているので、人間による解析ができない。しかし、クラスタ数の予想をし、その予想に最も近いパラメータを決めるようにしている。そもそもクラスタリングにおける「最適」という概念が客観的に決めかねるので、今後の研究に委ねるとする。

これらの課題を克服すれば、本研究の結果よりも遥に精度のいいシステムが出来上がると思われる。

謝辞

大学院に入ってから 2 年が経ちますが、研究室の皆さんにはご迷惑をかけています。就活の時は私が研究室に来られず、研究もあまりする時間が無かったのですが、暖かく見守って下さいました。特に木村さんは修論の研究では親切に話を聞いて下さり、いいアドバイスもたくさん頂きました。安達先生もいつも笑顔でとても親切でジェントルマンです。安達先生が私の身勝手をお赦しいただいたことには心底から感激です。なお、卒業された倉沢さんにもよくお世話になり、実に暖かく接してくれたことを覚えています。一緒に飲み会に行ったり、合宿に行ったりする記憶は鮮明に浮かんでいます。この研究室に入ってわがままな私を誰も責めないでいてくれたことに今でもすごく感謝しており、生涯に渡って皆様のご厚意を忘れることはありません。

また、研究が詰まった時に私を元気づけてくれた留学生の友達である何力、Yao Sen たちにも感謝しています。これから共に卒業して社会と向き合う仲間が一人でも多いことに越したことがありません。この修論で学生と別れを告げ、社会の海に帆を上げようではありませんか。

参考文献

[1] Pierre P. Senellart and Vincent D. Blondel
Automatic discovery of similar words
Proceedings of the 17th Annual International ACM-SIGIR chapter in: Survey of Text Mining

[2] Dekang Lin
Automatic Retrieval and Clustering of Similar Words
In Proceedings of COLING-ACL,1998.

[3] R.Ng and J.Han.
Efficient and Effective Clustering methods for spatial data mining
in VLDB-94,1994

[4] David Sánchez and Antonio Moreno
Automatic discovery of synonyms and lexicalizations from the Web
In Proceeding of the 2005 conference on Artificial Intelligence Research and Development

[5] Thomas K. Landauer, and Susan T. Dumais
A Solution to Plato's Problem: The Latent Semantic Analysis Theory of Acquisition, Induction, and Representation of Knowledge
In APA AMERICAN PSYCHOLOGICAL ASSOCIATION, Pages: 211-240

[6] Lonneke van der Plas and J ̈org Tiedemann
Finding synonyms using automatic word alignment and measures of distributional similarity
In Proceeding COLING-ACL '06 Proceedings of the COLING/ACL

[7] Olivier Ferret
Testing Semantic Similarity Measures for Extracting Synonyms from a Corpus
In Proceedings of the International Conference on Language Resources and Evaluation, LREC 2010

[8] Christian. B and Kjetil.N
Extracting Named Entities and Synonyms from Wikipedia
2010 24th IEEE International Conference on Advanced Information Networking and Applications

[9] Peter D. Turney
Mining the Web for Synonyms:PMI-IR versus LSA on TOEFL
In Institute for Information Technology, National Research Council of Canada,lecture notes in computer science

[10] Xing Wei, Fuchun Peng, Huishin Tseng, Yumao Lu, Xuerui Wang, Benoit Dumoulin
Search with Synonyms: Problems and Solutions
In Proceeding COLING '10 Proceedings of the 23rd International Conference on Computational Linguistics

[11] Thomas K. Landauer, Peter W. Foltz, Darrell Laham

An Introduction to Latent Semantic Analysis
In Discourse Processes 1998, Pages 259 - 284

[12] Hua-Ping Zhang ,Hong-Kui Yu, De-Yi Xiong and Qun Liu
HHMM-based Chinese Lexical Analyzer ICTCLAS
In Proceeding SIGHAN '03 Proceedings of the second SIGHAN workshop on Chinese language processing

[13] 木村友秋　藤井敦
評判情報の検索における隠語的造語法の応用
言語処理学会第15回年次大会発表論文

[14] W.-Y. Shieh, T.-F. Chen, J. J.-J. Shann, and C.-P.Chung
Similarity Based Chinese Synonym Collocation Extraction
International Journal of Computational Linguistics and Chinese Language Processing,2005

[15]Ido.Dagan, Lilian.Lee, and Fernando Pereira
Similarity-based methods for word sense disambiguation
In Proceeding ACL '98 Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics

[16] Tong Wang and Graeme Hirst
Extracting Synonyms from Dictionary Definitions
In Proceedings of the International Conference RANLP-2009

[17] Philippe Muller, Nabil Hathout, Bruno Gaume
Synonym extraction using a semantic distance on a dictionary
In Proceeding TextGraphs-1 Proceedings of the First Workshop on Graph Based Methods for Natural Language Processing

[18] Yunfeng Liu, Huan Qi, Jianmin Dai
中文信息的潜在语义分析
Journal of South China University of Technology Vol.32 November 2004

[19] Johanna Geiß
Latent semantic sentence clustering for multi-document summarization
Technical Report Number 802,UCAM-CL-TR-802

[20] Jen-Tzung Chien, Meng-Sung Wu, Hua-Jui Peng
Latent Semantic Language Modeling and Smoothing
In Computational Linguistics and Chinese Language Processing Vol.9 No.2 August 2004

[21] BRUNO GALMAR, JENN-YEU CHEN
Identifying Different Meanings of a Chinese Morpheme through Latent Semantic Analysis and Minimum Spanning Tree Analysis
IJCLA VOL. 1, NO. 1-2, JAN-DEC 2010, PP. 153-168
[22] Krister LIND´EN, Jussi PIITULAINEN
Discovering Synonyms and Other Related Words

CompuTerm 2004 - 3rd International Workshop on Computational Terminology

[23] Barbara Rosario
Latent Semantic Indexing: An overview
INFOSYS 240 Spring 2000 Final Paper

[24] Thomas Hofmann
Probabilistic Latent Semantic Analysis
UAI 99,Stockholm

[25] XIquan Yang,Na Sun,Tieli Sun,Xueya Cao,Xiaojuan Zheng
THE APPLICATION OF LATENT SEMANTIC INDEXING AND ONTOLOGY IN TEXT CLASSIFICATION
ICIC International Vol.5,Number 12,Dec 2009

[26] Minglei Chen,Xuecheng Wang ,Huawei Ke
Using Latent Semantic Analysis to create a Chinese Semantic Space and the validation of psychological reality
Chinese Journal of Psychology 2009, Vol. 51, No. 4

[27]中国インターネットユーザ数　http://www.chinanews.com/it/2011/07-19/3192864.shtml