

常識判断のための概念ベース構築法

小島 一秀 渡部 広一 河岡 司

同志社大学大学院工学研究科
〒610-0394 京都府京田辺市多々羅都谷 1-3

あらまし 常識判断メカニズムの中核となる機構は、概念ベースと概念の関連に基づく連想機能と考えられる。概念ベースは拡張、修正を容易にするため、できるだけ単純な構造が望ましい。現状の概念ベースでは概念を属性と出現頻度等から得られる重みで定義しているが、本稿では、自動精練を可能とするためこれを概念間に存在する明確な関係の種類により決定する方式を提案している。重み付けに使用した概念間の関係情報は、国語辞書から機械的に抽出した。評価には概念ベースを使って概念間の関係の深さを評価する関連度計算を使用した。この関連度による評価実験により、この重み付けをすると概念ベースから導かれる関連度が人間の感覚により近くなることを示した。

キーワード 常識判断 自動構築 国語辞書 概念ベース 関連度

Method of Concept-Base Construction for Commonsense Judgement

Kazuhide Kojima, Hirokazu Watabe, Tsukasa Kawaoka

Graduate School of Engineering, Doshisha University
Tanabe, Kyoto 610-0394

Abstract It is thought that the main elements of commonsense judgment like a human being are a concept-base and the association mechanism based on the association between concepts. It is expected that a structure of the concept-base is as simple as possible for ease concept-base expansion and correction. In the current concept-base concepts are defined by attributes and its weights which are decided by frequency. This paper proposes the method deciding the weight of attributes with clear relation types between concepts. The relation information deciding the weight of attributes is mechanically made from a language dictionary. Measuring the degree of association which uses the concept-base is used in evaluation the proposed method deciding the weight of attributes. It is shown that the degree of association calculated by the concept-base whose weights decided the proposed method is closer to human judge.

key words commonsense judgment, automated construction, language dictionary, concept-base, degree of association

1 はじめに

情報処理システムは、ここ数年を振り返ってみても従来にもまして急速な発展を続けており、人間社会のあらゆる分野で活用され、もはや欠かすことのできない存在になっている。しかし、これらの発展は、ほとんど、機能面、性能面での高度化に起因するものであり、知的な観点での著しい進展はみられない。今後、情報処理システムの機能・性能面での高度化が進めば進むほど、知的な情報処理へのあこがれは一層強まってくるものと思われる。“知的”や“知能”については古くから哲学、生物学、認知科学、理学、工学の幅広い領域で各種の研究が行われてきた。工学的な観点での“知的”に限っても、人工知能の実現可能性をめぐる多くの議論、研究がなされてきているが、未だに“知的”に関する明快な定義さえ与えられていない。

我々の研究のねらいは、このように難しく、やっかいな“知的”の本質の解明に挑戦しようとするものではなく、知的と呼んでもあまり違和感がなく、しかも、従来とは異なるメカニズムで、情報処理の高度化につながるような現実的なメカニズムを創出することにある。これまで、コンピュータ技術をベースに、より知的なメカニズムを求め、ゲーム理論や知識処理など人工知能と呼ばれる分野を中心に、自然言語処理、ファジィ、ニューラルネットワーク、遺伝的アルゴリズム、人工生命などの研究が行われ、それぞれに着実な成果を積み上げてきている。しかし、これらの多くは、基本的には、従来の情報処理技術とは比較的独立な技術として、あるいは、情報処理の特定の応用領域への適用を対象に研究が行われてきた。

本研究で対象とする“常識判断メカニズム”は、コンピュータシステムにおいて、オペレーティングシステム(OS)がハードウェアの効率的で使い易い利用メカニズムを汎用的に提供しているように、コンピュータシステムに“概念ベース”と呼ぶある種の知識をコンピュータ資源として持たせ、常識判断メカニズムとして、これの利用メカニズムを汎用的に提供することをねらうものである。データベース、ワードプロセッサ、帳票処理、インターネットなどの情報処理システム(プログラム)の開発において、OS機能と同様、これらの知的判断メカニズムを適宜に使うことにより、柔軟かいユーザインタフェースをもち、利用者の意図も汲み取ることのできる、“知的”と呼んでもよいような情報処理システムを開発することができると考えている。

概念ベースにおいて、概念はそれを説明する属性と呼ばれる概念とその重みで定義される。概念ベースは複数存在する。まず、複数の辞書の語義文から機械構築し、単語の出現頻度等で重みを付けた概念ベース(CB1)[1]がある。次に、その属性と重みの信頼性向上のために属性を重み順に30個選び出し、その重みを全て同じにした概念ベース(CB2)がある。

しかしながら、全ての属性がその概念の定義に同じだけ貢献しているとは考えられず、何らかの適切な重みの付け方が必要となる。

本稿では、このCB2を基に概念と属性の関係が異なるものかによって各属性の重みを適切に決定する方法を提案する。概念と属性の関係は、CB1, 2とは別に辞書から自動生成した概念ベース構築のための詳細な知識である“概念ソース”から得る。

続く2章では提案する重み付け方式により改良を行う概念ベースの内容と構造について述べる。3章では、概念ソースの内容と構築方法について述べ、この概念ソースを使った概念ベースの改良について4章で述べる。5章では、概念ベースの評価方法について述べ、6章では重みの付けの効果とその重みの適切な値を実験によって示す。

2 概念ベース

2.1 概念ベース CB1

概念ベースでは、概念 C を他の概念(属性)とその重み w_i の集合で定義する。

概念は次の概念ベクトル表現で表現できる。概念ベクトルの要素数は全概念数だけあり、各要素はその位置に対応づけられた概念(属性)の重みを示す。ベクトルの大きさは1に正規化しておく。

$$C = (w_1, w_2, \dots, w_m) \dots (1)$$

$$0 \leq w_i \leq 1$$

$$\sum_{i=1}^m w_i^2 = 1$$

各属性と重みは、複数の国語辞書等の語義文から自立語の出現頻度に基づいて獲得する。さらに、その自己参照による新たな属性の追加、不要な属性の統計的な除去から成る精錬を行うことによって完全に機械構築している[1]。

概念数は約4万、属性数は約150万となっている。

2.2 概念ベース CB2

CB2は、CB1の属性を30以下に打ち切り、属性の重みを全て1にした概念ベースである[2]。概念の属性が30より多い場合、重みの上位から30個のみを残し、30以下の場合打ち切りはない。

ベクトルで表現すると、0以外の要素が30個以下のベクトルとなる。

概念数は約4万、属性数は約90万となっている。

3 概念ソース

3.1 概念ソースの内容

概念ソースは、概念ベース構築のための概念に関する詳細な情報で、概念ベースとは別に国語辞書から機械構築している[3]。例を表1に示す。概念ソースに入っている情報はその概念の綴り、概念を説明する単語の綴り(関連語)とその関連語と概念の関係(関係型)である。

基本的には、概念ソースの一つの概念は構築元辞書の一つの見出し語に対応する。しかし、見出し語の意味が分類されている場合は、その最も細かい一分類と対応する。したがって、概念ソースの概念数は、辞書の見出し語より多い。

概念ソースの関連語は概念ベースの属性に相当するが、次の点で異なる。属性は、概念ベース中の一つの概念を示す。したがって、概念の属性から、さらにその属性も取り出せる。一方、概念ソースの関連語は、概念の綴りである。したがって、概念ソース中のどの概念により定義されているのかを、ある程度は限定できても特定することはできない。関係型は同義、類義、上位、対義、尊敬、丁寧、英字綴りの8種類となっている。また、関連型には誤りも存在し、表1に示す概念“藍色”の関連語“中間”はその例である。

情報量は、概念数16万、関連語数100万、一概念の持つ平均関連語数6となっている。また、各関係型を持つ関連語数は表2に示す。

表1:概念ソースの例

綴り	関係型	関連語
アートペーパー	なし	表面
	なし	なめらかだ
	なし	つや
	なし	上質だ
	上位	紙
	同義	アート紙
	同義	アート
	英字	artpaper
あいいろ	なし	濃い
藍色	上位	青色
	なし	紺
	なし	色
	なし	青色
	上位	中間
	同義	インジゴ
	同義	あい

表2:関係型の数

関係型	関連語数
なし	844,080
同義	60,028
類義	22,303
上位	78,997
対義	6,589
尊敬	587
丁寧	13
英字	8,417

3.2 概念ソースの構築方法

ここでは、概念ソースの構築方法を図1の処理を例に述べる。

概念ソースの構築は、辞書の規則的な表記構造から情報を得る構造情報、自然言語から情報を得る自然言語内処理、これらの処理で得た情報をまとめ概念ソースを構築する統合処理の三段階からなる。

アメジスト[1][鉞]宝石の一つ。紫色の水晶。紫水晶。(参)二月の誕生石。[2]紫色。=アメジスト。▽amethyst

図1:辞書の説明文の例

構造処理では、辞書の規則的な表記構造の利用して、コンパイラと同様の処理を行う。処理は要素分解、構文解析木生成、構造情報生成の三段階からなる。処理結果の例を表3に示す。

要素分解はコンパイラの字句解析に対応し文字列を意味のある最小要素に分解する。ただし、説明文の自然言語部分はひとつかたまりの要素として扱う。次の構文解析木生成はコンパイラの構文解析に対応する。辞書の表記構造規則にしたがって解析を行う。最後は、構造情報生成である。これはコンパイラのコード生成に対応する。構文解析木を解読しながら、分類や関係に関する情報の付いた単位文を生成する。単位文とは、説明文の自然言語部分を区点で区切った文字列で、多くの場合は通常の文である。分類や関係に関する情報は、辞書の記号から得ている。

表3:構造処理の例

分類	関係	単位文
1	なし	宝石の一つ。
1	なし	紫色の水晶。
1	なし	紫水晶。
1	なし	二月の誕生石。
2	なし	紫色。
2	なし	アメジスト。
2	英字	amethyst

自然言語内処理は、辞書の統一された定型的な表現を利用し、形態素解析と簡単なパターン処理で自然言語から情報を取り出す。処理は、文字検出、形態素解析、形態素検出の三段階からなる。処理結果の例を表4に示す。

文字検出、形態素検出では情報文を検出する。この情報文とは、関連語以外にも情報が得られる単位文である。文字検出は、文字の検査により情報文を検出する。表4では、一つ前の単語が見出しの上位である“の一つ。”を検出している。形態素検出は、品詞の配列により情報文を検出する。表4では、次の二種類を検出している。名詞止めからは、その名詞が見出し語の上位であることがわかり、全て名詞の単位文は一つの単語であり、見出し語と同義であることがわかる。文字検出、形態素検出で使用する規則は、辞書を調査して決定した。

形態素解析[4]は、単位文を単語に分割しその品詞を決定する処理である。ただし、構造処理で関係等が得られた単位文は自然言語内処理の対象外としている(表4のamethyst)。

最後が統合処理である。ここでは、構造処理、自然言語内処理で得た情報をとりまとめ概念ベースを構築する。結果の例を表5に示す。

表 4: 自然言語内処理の例(“/”は形態素の区切り)

分類	文字検出	形態素検出	単位文
1	“の一つ”		宝石/の/一つ/。
1		名詞止め	紫色/の/水晶。
1		全て名詞	紫水晶/。
1		名詞止め	二月/の/誕生石。
2		全て名詞	紫色/。
2		全て名詞	アメジスト/。
2	-	-	amethyst

表 6: CB3 の重みの数

関係型	重みの数
同義	11223
類義	4719
反対	502
上位	19542

表 5: 統合処理の例

分類	関係型	単位文
1	上位	宝石
1	なし	紫色
1	上位	水晶
1	同義	紫水晶
1	なし	二月
1	上位	誕生石
2	同義	紫色
2	同義	アメジスト
2	英字	amethyst

4 概念ベース CB2 の改良

概念ベースの改良は、CB2 の属性と、概念ソースの関連語を比較し、属性と関連語が一致したとき、関連語の関係型に応じた重みを属性に付けることにより行う。

重みは、属性がどの関係型であるかによって、5 種類の値を付けた。関係型のない属性の重みには、基準値 1.0 を設定する。尊敬、丁寧、英字は意味的に同義に近いので、同義とみなしている。重みの具体的な値は、実験において様々な値を試した。

改良された概念ベース(CB3)の属性で 1.0 以外の重みが付いたのは、全属性 90 万の 4%にあたる 3 万 6 千属性である。CB3 における各関係型の重みの数(表 6)は、上位が最も多く、次いで同義、類義、反対となっている。反対はそれ以外と比べると数が少ないが、各関係型の数は概念ソースの関係型の数にほぼ対応している。CB3 の一部を表 7 に示す。

CB2 の概念と概念ソースの概念は綴りを通して対応させるが、CB2、概念ソースともに概念と綴りの対応は 1 対 1 ではない。このため、CB2 の概念と概念ソースの概念とが全く異なった概念に対応づけられる可能性がある。しかし、全く異なる概念の属性が一致することは起こりにくいため、悪影響は少ないと考えている。

改良の手順(図 2)は次のようになっている。まず、CB2 の概念 C_B に対応した、概念ソースの概念を選ぶ。このとき、 C_B の綴りは複数あるがその全てを使い、綴りに合う概念ソース側の概念全てを対象とする。そして、 C_B の属性の綴りと概念ソースの概念 C_S の関連語を照合する。このとき一致すれば、 C_S の関連語の関係型に応じた重みを C_B の属性に付ける。

```
//CB2 概念ループ
for i1=1 to cmax
{
//CB2 概念の綴りループ
for i2=1 to cbspell_max(i1)
{
//CS 概念ループ
for i3=1 to cscon_max( cbspell(i1, i2) )
{
//CS 概念の関連語ループ
for i4=1 to csword_max( cbspell(i1, i2) )
{
//CB2 概念の属性ループ
for i5=1 to cbatt_max(i1)
{
//CB2 概念の属性の綴りループ
for i6=1 to cbspell_max( cbatt(i1, i5) )
{
if cbspell( cbatt(i1, i5) )
=csword( cscon( cbspell(i1, i2), i3 ), i4 )
then
cbwrite( i1, i5,
csrelation( cscon( cbspell(i1, i2),
i3 ), i4 ) )
}
}
}
}
}
}
}
}

i1, ..., i6:カウンタ
【CB2 アクセス関数】
cbspell(a1, a2): 概念 a1 の a2 番目の綴りを返す
cbspell_max(a1): 概念 a1 の綴り数を返す
cbatt(a1, a2): 概念 a1 の a2 番目の属性が持つ概念番号を返す
cbatt_max(a1): 概念 a1 の属性数を返す
write(a1, a2, a3): 概念 a1 の a2 番目の属性に、関係型 a3 の重みを書き込む
【概念ソースアクセス関数】
cscon(a1, a2): 綴りが a1 である a2 番目の概念の概念番号を返す
cson_max(a1): 綴りが a1 である概念数
csword(a1, a2): 概念 a1 の a2 番目の関連語を返す。
csrelation(a1, a2): 概念 a1 の a2 番目の関係型を返す。
csword_max(a1): 綴りが a1 の関連語数
ここでは、CB2、概念ソースともに概念を 1 から最大概念数の数値で指定している。
```

図 2: CB2 の概念ソースによる改良アルゴリズム

表 7: CB3 の一部一概念: 太陽

関係型	属性	関係型	属性
なし	太陽系	なし	高山
なし	地球	なし	牛
なし	太陽	なし	平均
上位	恒星	類義	日
なし	中心	なし	希望
なし	光	なし	日月
なし	日食	なし	放つ
なし	天体	なし	水星
なし	惑星	なし	木星
なし	文芸	なし	距離
同義	日輪	なし	冬至
なし	半径	なし	彗星
類義	天日	なし	熱
なし	温度	なし	外側
なし	廃刊	なし	金環食

表 8: 評価尺度の例

M _x	M _A	M _B	M _C
樹木	木	木の葉	頭
天気	天候	雨	写真
時刻	時間	時計	消しごむ
海	海洋	波	耳
瞳	目	顔	靴
人	人間	動物	箱
子供	童	大人	雲
辞書	辞典	本	家

5 概念ベースの評価

5.1 評価方法

CB3 の評価には、概念間の関係の深さを評価する関連度を利用する。この関連度の計算では概念ベースを使うため、計算結果が概念ベースの性能の指標となる。

まず、ある特定の関係にある概念の組を集めた評価尺度を準備する。そして、評価したい概念ベースを使い、評価尺度の概念に対して関連度の計算を行う。次に、関連度計算の結果を評価するための評価指数を計算する。そして、評価指数を使って概念ベースの評価を行う。

5.2 試行した属性の重み

重みは 5 種類あるが、関係型なしの重みを固定し、残る 4 つの重みを変化させて関連度計算を行う。これは、関連度の計算時に重みが正規化されるためである。

重みは、基準値を 1.0 として 0.5, 1.0, 1.5, 2.0, 2.5, 3.0 の 6 種類の値を試行した。4 変数に対し 6 種類の値を付けるため全部で $6^4=1296$ 通りの試行となる。

5.3 評価尺度

評価尺度とは、人手で作成した 1 セットが 4 つの概念からなるデータである(表 8)。4 つの概念のうち 1 つは、そのセットの基準となる概念 M_x である。残り 3 つは、概念 M_x と同義または類義の概念 M_A 、関係のある概念 M_B 、関係のない概念 M_C である。今回使った評価尺度には、このような概念が 500 セット入っている。

M_x と M_A , M_B , M_C それぞれの関連度をそれぞれ r_A, r_B, r_C とし、評価尺度全体における関連度の平均を R_A, R_B, R_C 、標準偏差を $\sigma_A, \sigma_B, \sigma_C$ とする。

常に $r_A > r_B > r_C$ となり、 r_A が 1 に近く、 r_C が 0 に近ければ、関連度計算の理想的な結果といえる。

5.4 関連度計算

関連度計算には、ベクトル内積関連度計算[1]を使用した。これは、対象概念の関連度ベクトルを正規化しその内積を関連度 R とする関連度計算法である。

$$R = C_A \cdot C_B \quad \dots (2)$$

ただし、 C_A, C_B は関連度計算を行う 2 つの概念ベクトルで、それぞれ行ベクトルと列ベクトルである。

5.5 評価指数

概念ベースに要求される特性を次の 2 点と仮定し評価指数を定義する[5]。

- (1) 関係の深い概念間の関連度と全く関係のない概念間の関連度の差が大きい。
- (2) 二つの候補概念があったとき、どちらが対象概念と関連が深いかを識別可能である。

まず、特性(1)の評価指数 F_1 を次のように定義する。

$$F_1 = (R_A - R_C) / (\sigma_A + \sigma_C) \quad \dots (3)$$

これは、 M_x と強く関連する概念 M_A と全く関連しない概念 M_C との平均関連度の差を標準偏差により正規化した指数である。

次に、特性(2)の評価指数 F_2 を次のように定義する。

$$F_2 = 1 / (1 + wg) \quad \dots (4)$$

ただし、 wg は評価尺度 100 セットあたりの $r_A \leq r_B$ となつた数である。

F_1, F_2 とも大きいほど良いので、関連度計算の結果の評価指数を次のように定義する。

$$F = F_1 \times F_2 \quad \dots (5)$$

したがって、概念ベースの質が良いほど F は大きい。

6 結果と考察

6.1 重み付けの効果

重み付けの効果を見るために、CB1, CB2, CB3 を使って評価尺度のベクトル内積関連度計算を行い、評価指数を計算した。CB3 では様々な重みを試行したが、ここではその中で最も高い F を採用する。

結果(図 3)を見ると、CB3 の F が最も高く、CB1 とほぼ同じくらいである。CB3 は CB2 を改良した概念ベースであるので、概念ベース改良の効果は F において 43%の改善と言える。

また、CB3 と CB1 の F はほぼ同であるため、CB3 の重みは、CB1 の辞書における概念の出現頻度などを使った重みと、CB2 を CB1 から生成するときに削除した属性分に相当すると言える。なお、CB1, CB3 の全属性数はそれぞれ 150 万と 90 万である。

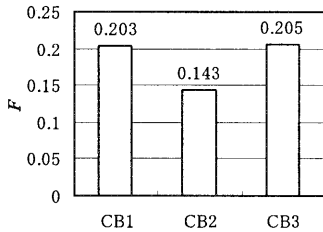


図 3: CB1, 2, 3 の評価指数

評価尺度中の重みの分布について見てみる。表 9 は、評価尺度中の概念が持つ属性の関係型の数と比率である。ただし、概念の重複を削除したデータである。表 9 からわかるように、評価尺度中の概念が持つ属性の中で、1.0 以外の重みが付いたのは 7% である。したがって、この改良方式による F の改善は、7%の属性が引き起こしていると言える。

重みが 1.0 以外の属性の比率は、概念ベースより、評価尺度の方が高い。これは、評価尺度に人間が入力するほど知られている概念は、辞書においても記述が詳細なためではないかと考える。

表 9: 評価尺度中の概念における属性の各関係型の数と比率(重複概念は削除)

関係型	数	比率(%)
なし	2285	93.0
同義	798	2.4
類義	532	1.6
上位	921	2.8
反対	34	0.1

6.2 適切な重み付け

6.1 で重みの効果を示したが、ここでは、どのような重み付けが適切であるかについて述べる。5.1 で述べたように、CB3 の重み付けは $6^4=1296$ 通り試行した。以降では、それらの評価値と各関係型の重みの関係を見ながら、適切な重みの付け方について述べる。

1296 通りの重みを、そのときの F によって 0.1 区切りでクラス分けし、各クラスにおける関係型に付けられた重みの比率を調べた。ただし、 F の値の区分によってクラス分けしたため、各クラスのサンプル数(表 10)は一律でない。 F の最大最小はそれぞれ 0.205, 0.121 である。試行した結果の上位 5 データを表 11 に示す。

表 10: 各クラスのサンプル数

クラス	F の値 (未満-以上)	サンプル数
1	0.21-0.20	33
2	0.20-0.19	210
3	0.19-0.18	336
4	0.18-0.17	204
5	0.17-0.16	158
6	0.16-0.15	156
7	0.15-0.14	129
8	0.14-0.13	34
9	0.13-0.12	36

表 11: F の上位 5 つの重み

同義	類義	上位	反対	F
1.5	2.5	2.0	2.0	0.205
1.5	2.5	2.0	2.5	0.205
1.5	2.5	2.0	1.5	0.205
1.5	2.5	2.0	3.0	0.205
1.5	3.0	2.0	2.0	0.204

同義の重みと F の関係は図 4 の通りである。このグラフでは、適切な重みでは右上がり、不適切な重みでは右下がりとなるはずである。したがって、同義の適切な重みは、1.5 以上と言える。クラス 9 から 1 にかけて 1.5 以上の重みの比率は同じように推移している。クラス 9 において、重み 2.0 が 0% であるが、クラス 9 のサンプル数は他のクラスより少ないためであると考えられる。クラス 8 から 1 の範囲で 1.5 以上の全ての重みにおいてほぼ右上がり傾向である。逆に、不適切な重みは 0.5 と 1.0 である。重み 0.5 は右下がりであり、最も F の低いクラス 1 では 100% である。重み 1.0 はクラス 1 で 0% であるが、少ないサンプル数のためと考える。以上より、同義の重みは、1.5 以上が適切で、1.0 以下が不適切であると考える。

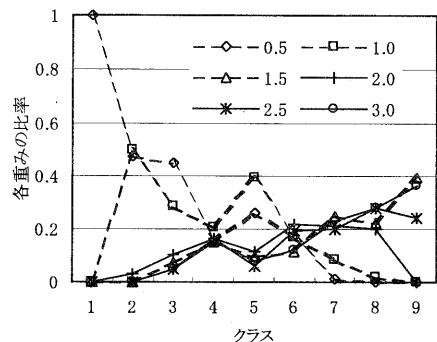


図 4: クラスと同義の各重みの比率

類義の重みと F の関係は図 5 の通りである。グラフから、適切な重みは 2.0, 2.5 と考える。どちらのグラフも、右上がりの傾向があり、サンプル数の少ないクラス 9 を除いても、その傾向は変わらない。クラス 9 の 2.5 の比率が 5 割以上と顕著である。これは、サンプル数が少ないためと考える。しかし、重み 3.0 の結果が 2.0, 2.5 よりふるわないことを考慮に入れると、適切な重みのピークがこの辺りにある可能性もある。逆に不適切な重みは見いだせない。2.0, 2.5 以外の重みは全てあまり傾向がない。全体に、同義ほど顕著なデータが出ないのは、類義の数が少ないためであろう。以上より、類義に適した重みは 2.0, 2.5 であると考える。

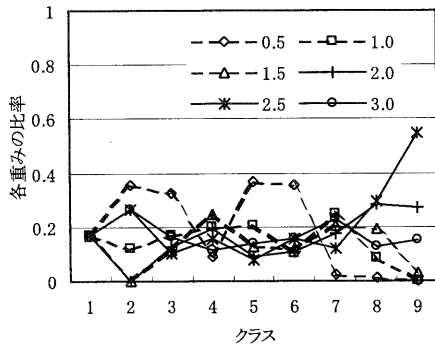


図 5: クラスと類義の各重みの比率

上位の重みと F の関係は図 6 の通りである。このグラフから適切な重みは 1.5, 2.0 と読みとれる。どちらも、右上がりの傾向である。また、クラス 9 における 2.0 の比率が 8 割以上と顕著に高いが、これには少ないサンプル数の影響があると考える。しかし、類義の場合同様 2.0 より大きい 2.5, 3.0 は 1.5, 2.0 と比べると結果がふるわないため、適切な重みのピークがこの辺りにある可能性もある。不適切な重みは 0.5 である。サンプル数の少ないクラス 1 を除いても右下がりの傾向が出ている。以上より、上位の適切な値は、1.5, 2.0 と考える。

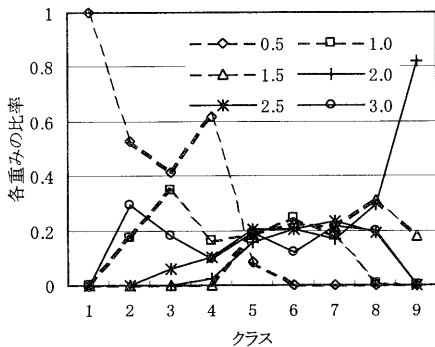


図 6: クラスと上位の各重みの比率

反対の重みと F の関係は図 7 の通りである。このグラフから適切な重み不適切な重みともに読みとれない。これは、反対の重みが 34 と他の重みより少ないためと考える。ただ、信頼性は低いクラス 9 だけ見ると完全に重みの順になっている。以上より、反対の重みはこの実験では決められないと考える。

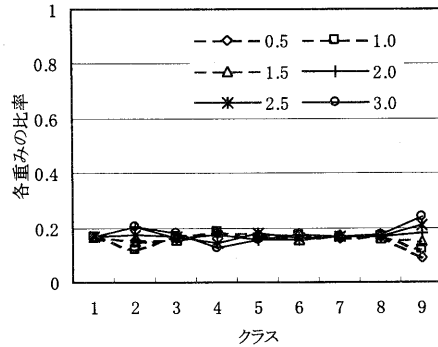


図 7: クラスと反対の各重みの比率

以上の関係型による重みの適不適を表 12 にまとめた。表から各関係型にはそれぞれ、特定の範囲に適切な重みがあることがわかる。また、サンプル数の少ない反対以外は、どれも 1 より大きな重みを付けて良いことがわかった。今回の重みの試行範囲では、同義の重みの限度はわからなかった。また、同義は 1.0 が不適で 1.5 が適切であるところから、他の関係型より重みの適不適に敏感なようである。表 13 にクラス 1 のデータを示しておく。

表 12: 各関係型の重みの適不適
(○適切, △どちらでもない, ×不適, -不明)

	0.5	1.0	1.5	2.0	2.5	3.0
同義	×	×	○	○	○	○
類義	△	△	△	○	○	△
上位	×	△	○	○	△	△
反対	-	-	-	-	-	-

7 おわりに

概念ベースを常識判断メカニズムの中心的な要素と考え、概念ベースの属性の重みを、同義、類義などの明確な概念間の関係によって決定する方式を提案した。提案方式の重み付けに使用した概念間の関係が辞書から機械的に取得可能で、人間から見ても比較的明確であり、種類も少ない。このため、概念ベースの構造は自動精練等のために単純さが要求されるが、提案方式はこの要求に合っていると考える。提案方式の有効性については概念ベースを使い概念間の関係を評価する関連度計算を利用して示した。

参考文献

- [1]笠原 要, 松澤 和光, 石川 勉:国語辞書を利用した日常語の類似性判別, 情報処理学会論文誌, Vol.38, No.7, pp.1272-1283(1997)
- [2]知的判断メカニズムのための概念間の類似度評価モデル:入江 毅, 渡部 広一, 河岡 司, 松澤 和光, 信学技報, Vol.98, No.499, AI98-75(1999)
- [3]小島 一秀, 渡部 広一, 河岡 司:電子化国語辞書を用いた概念ベース自動構築における前処理の自動化, 情報処理学会第 58 回全国大会, 1M-6, (1999)
- [4]<http://cactus.aist-nara.ac.jp/lab/nlt/chasen.html>
- [5]石川 勉, 井澤 潤次郎, 笠原 要:単語の意味に関する概念ベースの類似性判断能力からの最適構成, 人工知能学会誌, Vol.13, No.3, pp.470-479(1999)

表 13:クラス1の重みとF

同義	類義	上位	反対	F
1.5	2.5	2.0	2.0	0.205
1.5	2.5	2.0	2.5	0.205
1.5	2.5	2.0	1.5	0.205
1.5	2.5	2.0	3.0	0.205
1.5	3.0	2.0	2.0	0.204
1.5	3.0	2.0	2.5	0.204
1.5	3.0	2.0	1.5	0.204
1.5	3.0	2.0	3.0	0.204
1.5	3.0	2.0	1.0	0.204
2.5	2.5	2.0	2.0	0.203
2.5	2.5	2.0	2.5	0.203
2.5	2.5	2.0	1.5	0.203
2.5	2.5	2.0	1.0	0.203
2.5	2.5	2.0	3.0	0.203
2.5	2.5	2.0	0.5	0.203
3.0	2.5	2.0	2.0	0.203
3.0	2.5	2.0	1.5	0.203
3.0	2.5	2.0	2.5	0.203
3.0	2.5	2.0	1.0	0.203
3.0	2.5	2.0	3.0	0.203
3.0	2.5	2.0	0.5	0.203
3.0	2.0	2.0	2.0	0.203
3.0	2.0	2.0	1.5	0.203
3.0	2.0	2.0	2.5	0.203
3.0	2.0	2.0	1.0	0.203
3.0	2.0	2.0	3.0	0.203
3.0	2.0	2.0	0.5	0.203
1.5	2.0	1.5	2.0	0.202
1.5	2.0	1.5	2.5	0.202
1.5	2.0	1.5	3.0	0.202
2.5	2.5	1.5	2.5	0.201
2.5	2.5	1.5	3.0	0.201
1.5	1.5	1.5	3.0	0.201