

Analysis of Strategic Knowledge in Back of the Envelope Reasoning

Praveen K. Paritosh and Kenneth D. Forbus

In Proceedings of the 20th National Conference on Artificial Intelligence (AAAI-05), Pittsburgh, PA. Vol.2, pp. 651-656, 2005

Abstract

Back of the envelope (BotE) reasoning とは、正確なデータおよびモデルが利用不可能な状況において、量的な答えを生成することである。またこのとき、利用可能なデータはしばしば不完全であり、一貫性もないことが多い。

概算は素早く行われ、非実用的かつ不可能であるかもしれないような詳細な解析よりも価値があり有用であるとされる。これは多大な時間や情報、他のリソースを必要としないためである。これらの推論は、日常的な状況に対しての常識 *common-sense* 推論の主要な要素である。

そこで我々は、「アメリカで 1 年間に健康に関して使われるお金はいくらか?」といった、12 の概算問題を解決するシステム、BotE-Solver を提案する。これは、異なる領域間を跨ぐ戦略を用いたり、Cyc knowledge base のライブラリを使用したりすることで問題解決を行うシステムである。

BotE-Solver は、汎用の問題解決フレームワークであり、*suggestion* として表わされる戦略を使用し、AND/OR 木による問題解決過程の追従を行う。この論文の重要な点は、知識レベルによる分析を行っているという点である。

我々は、以上のような広範囲の問題解決を行うためのコアとなる 7 つの推定に関する戦略を示す。我々は、この 7 つが Back of the envelope 問題を解決するためのすべての戦略であると仮定する。

以下の 2 つの分析は、この仮説を支持するものである。

1) Clifford Swartz の "Back-of-the-Envelope Physics" [Swartz 2003] にまとめられている、物理学の問題(n=44)に対する実証的分析

2) BotE-Solver の使用した戦略の分析

1. Introduction

次の例を考えてみていただきたい:

- ・アメリカには何人の K-8 学校(Kindergarden through the 8th grade)の教師がいるか?
- ・アメリカで 1 年に健康に関して使われるお金はいくらか?
- ・今シーズン 1 試合当たりのジェイソン・キッドの得点はどのくらいか?

これらの問題に共通にあるものは次のとおりである

- 1) 数値の答えを求めるものであること
- 2) 正確な答えが見つかることは難しいかもしれないが、十分によい概算を行うことは可能であること

この論文では、我々は BotE-Solver (上記の問題を含む 12 の問題を解決することができるシステム)を提案する。

上記のような推論には 2 つの重要な部分がある。複雑な問題を単純化するためにヒューリスティックや戦略を使用することと、適切な数の推定を行うために数の感覚を使用することの 2 つである。

BotE-Solver の中心には、推定に用いる戦略のライブラリがある。

この研究の 1 つのゴールは、強力で再使用可能な、コアとなるいくつかの戦略を見つけることであった。結論として、我々は 7 つの戦略のみでほとんどの推論をとらえることができると主張する。

2. Analysis of BotE Strategies

すべての BotE 概算問題は、あるパラメータの量的な数値を求めるものである。

我々のアプローチでは、問題解決を 2 つのプロセスに分割する。

1. Strategies:

現在の問題を、より容易な問題に変換するために戦略を使用すること。

2. Estimation:

あるパラメーターに対して、数値的な推定を行うこと。既に数を知っていることがあるかもしれないし、推定を行うために類似の例を使用するかもしれない

まずは、問題を形式化することから始める。

BotE 問題を抽象的に表すため、 Q が量、 O がオブジェクト、そして $?V$ が求められる未知の数値であるとする、 $(Q \ O \ ?V)$ と表現できる。

例えば、「米国に何人の K-8 の教師がいるか？」という問題のとき、 Q は要素の数（基数）で、 O は米国にいる K-8 教師の集合である。

ただし、問題を量とオブジェクトのペアとして表す際、必ずしも 1 通りには定まるとは限らない。「アメリカでは 1 年あたりいくら新聞にお金が使われるか？」といった問題は、次の 2 通りに表現することができる。:

Q =費用、 O =米国で売られた年間すべての新聞

Q =1 年間の費用、 O =米国のすべての新聞

問題を $(Q \ O \ ?V)$ と表わすと、戦略はそれを別の問題 $\{(Q_i \ O_i \ ?V_i)\}$ に変換する。

V_i は既に知っている値であるか、あるいは推定することが始めより容易なものである。

問題の変換に加えて、それぞれの戦略は、次の質問に対する答えを含んでいる

- ・それはいつ適用されるか？
- ・ $?V$ を見つけるために、どのように $?V_i$ を結合させるか
- ・ $?V$ の推定に対する確信はどうであるか。

上記のように戦略を表わすと、どのように変換されるかに基づいた、3つの大きな戦略パターンがあることが確認できる:

1. Object-based: $(Q \ O \ ?V) \rightarrow \{(Q \ O_i \ ?V_i)\}$
2. Quantity-based: $(Q \ O \ ?V) \rightarrow \{(Q_i \ O \ ?V_i)\}$
3. System-based: $(Q \ O \ ?V) \rightarrow \{(Q_i \ O_i \ ?V_i)\}$

次に、S1 から S7 と番号を付けられた 7 つの戦略を示し、それぞれについて順番に見ていく。

<Object-based Strategies> オブジェクトベース戦略

S1~S3

S1. Mereology:

この戦略は、あるオブジェクトを各部分のオブジェクトに変換(分割)するものである。

例えば、果物が乗っているかごの重量は、すべての果物の重量とかごの重量を合計したものである。

O が同質のオブジェクトであるとき、総体のオブジェクトは部分に分けることができる。
この戦略を適用するには、部分は重複してはならない。

S2. Similarity:

この戦略は、オブジェクトを類似した他のオブジェクトに変換するものである。
例えば、オーストリアの人口を求める際、2つの国の類似性に基づいて、スイスの人口から求めるのは理にかなった推定となりうる。

この戦略は2つの方法で実践することができる:

- ・ O is similar to O1 $\Rightarrow ?V=?V1$
- ・ O is similar to {O_i} $\Rightarrow ?V=Average(?V_i)$

2つのオブジェクトが類似している場合でも、2つのオブジェクトに対するすべての量の値が同じ推論が行えるという保証はない。

S3. Ontology:

この戦略は、問題の量を推定するために用いられ、存在論階層 ontology hierarchy から他のオブジェクトを見つけようとするものである。

最もシンプルなものとしては、O が O1 の実例である場合、私たちは、実例の値を推測するために、そのカテゴリに関する知識を使用することができる。

例えば、ジェイソン・キッドがポイントガードであるということを知っていれば、彼の身長を推定するにあたって、「ポイントガードはチームの他の選手より比較的背が低い」という知識を使用することができる。もしもポイントガードに関する情報を持っていなければ、ジェイソン・キッドの身長の推定において、彼がバスケットボール選手であるという事実を使用することができるだろう。

また、上の階層に行けば行くほど、推定の正確性は下がっていく。

(例)ある特定の椅子→肘掛け椅子→椅子→家具

<Quantity-based Strategies> 数量ベース戦略

S4~S5

S4. Density:

この戦略は、ある量を密度量(電束密度、人口密度など)および範囲量に変換するものである。
学生1人当たりの教師数といった量も密度であると考えられる。

S5. Domain Laws:

この戦略は、領域固有の法則を用いることで、ある量に関連する他の量に変換するものである。領域の法則というのは、経験則と同様に物理学の法則なども含む。

例えば、ニュートンの第 2 の運動の法則($F=m*a$)は、オブジェクトの力に質量や加速度を関連づけている。

他にも、幾何学、微積分、代数学などの法則が適用できる。

<System-based Strategies> システムベース戦略

S6~S7

システムベース戦略は、量とオブジェクトの双方を別の量、オブジェクトに変換するものである。

S6. System Laws:

このクラスは、システム全体に適用可能な物理法則から成っている。

多くの物理量 (例えばエネルギー、運動量、角運動量など) が保存則により保持される。そのため、システムの 2 つの異なる状態の量の値についてバランス方程式を書くことができる。このような、熱エネルギーから運動エネルギーなどの変換のことを指す。

S7.Scale-up:

これは多くの場合、実践的な BotE 戦略である。

ある狭いスケールで成り立つ物理法則のモデルを、フルスケールのモデルに対する量の値を推定するために使用すること戦略である。

例えば、レイノルズ数は、流体の性質を持つ数である。このレイノルズ数をスケールアップすることで流体モデル全般に適用することなどが挙げられる

3. Discussion

前述の 7 つの戦略が広い適用範囲を持っていることを確認するために、Clifford Swartz の “Back-of-the-envelope Physics.” の問題を使用した。

この本は、物理学の様々な領域から概算問題を集めたものである。力学や圧力、熱力学、天文学などの問題をはじめ、すべての問題(n=44)について分析した。

Table I に分析の結果を示す。

(注 Swartz の本の問題と、BotE-Solver の解いた問題は異なっている)

Table I: Distribution of strategies.

Source →		Swartz		BotE-Solver	
Strategy ↓		Number of times used.	% of times used	Number of times used.	% of times used
S1	Mereology	11	14	9	32
S2	Similarity	5	6	5	18
S3	Ontology	6	8	0	0
S4	Density	10	13	10	36
S5	Domain laws	29	37	3	11
S6	System laws	11	14	1	4
S7	Scale-up	2	3	0	0
	Other	5	6	0	0
	Total	79	100	28	100

適用範囲について:

S1～S7の7つの戦略は、Swartzの本の問題で使用した戦略の94%を占めていた。「その他」カテゴリーに分類された戦略は、5つのみであった。

これは、7つの戦略のみでほとんどのパターンを網羅できるということである。

領域固有性について:

S1からS4の戦略は、領域に依存しない戦略であり、全体でおよそ40%使用された戦略であった。残り60%の戦略(S5～S7)は、領域固有の戦略であり、domain laws戦略が全体で最大の割合を占めていた。

興味深いことに、BotE-Solverでは、領域固有の戦略の割合が15%のみであった。これは、ほとんどの問題が物理学の法則を適用するSwartzの本と比べ、常識 common-sense な領域の問題を解いたからであると考えられる。

上記の分析の後、我々は問題解決プログラムを実装した。問題解決プログラム中の戦略知識をより簡潔な表現に設計するガイドとして先の分析を利用している。

次のセクションでは、BotE-Solverの現在の実装部について短い記述を示す。

4. Implementation of BotE-Solver

問題解決のコンピュータ計算モデルは、問題の表現を形式化しなければならない。また、領域知識にアクセスし、適切な知識を検索する能力も必要とされる。そして、問題が複雑で答えが直接見つからない場合には戦略を用いる必要があるし、どこまで解いたのかといった進捗状況を追従するワークスペースを保持するということも必要になってくる。

これに対し、我々は以下の3つを用いて BotE-Solver を実装した。

- 領域知識が蓄えられた巨大な知識ベース
- 戦略の表現としての提案 (suggestion)
- ワークスペースを保持するためのモデルとしての AND/OR 木

次に、コアとなるアルゴリズムを示していく。

• 領域知識

知識ベース Knowledge Base (KB) および推論エンジンは、この研究を支える基盤の一部である。我々の知識ベースのコンテンツは、Cycorp による 120 万個の Cyc knowledge base のサブセットを使用している。それらは、日常における様々なオブジェクトや人、出来事や関係性に関する表現を提供する。問題、解答、戦略はすべて一様に表わされ、この KB に蓄えられる。

また、ノースウェスタン大学およびゼロックス PARC と共同で開発されている推論エンジン FIRE を使用した。

• 戦略

私たちは suggestion を使用して、戦略を実行可能なプログラムにする。Suggestion とは、戦略をより具体化したものであり、問題を分解する役割を持つ。

```
(defSuggestion HouseholdStrategyForCountingUnits
  :trigger
    (unitsTotal ?obj ?place ?time ?total-units)
  :test (ownedBy ?obj FamilyCohabitationUnit)
  :subgoals
    ((numberOfHouseholds ?place ?time
      ?num-households)
     (unitsPerHousehold ?obj
      ?units-per-household))
  :result-step (evaluate ?total-units
    (TimesFn ?num-households
      ?units-per-household)))
Figure 1. An example suggestion.
```

Figure 1 は、HouseholdStrategy-ForCountingUnits は、自動車の数を推定するために、家族の世帯の数を見つけることにより解決しようとしたという状況である。

世帯(KB 中の FamilyCohabitationUnit として参照されたデータ)ごとに何か(ここでは自動車)が所有されるということを知っていれば、

世帯の数×1つの世帯当たりのユニット(自動車)数

から、全体でどれだけのユニットが所有されているのか知ることができる。

表中にある以下の部分は、1つの suggestion のうちの4つの部分である

・ Trigger トリガー ・ Test テスト ・ Subgoals サブゴール ・ Result-step 結論ステップ

・問題解決過程の追従

問題解決過程を追従するため、BotE-Solver は AND/OR 木を使用している。AND/OR 木と我々の表現の間のマッピングは非常に直接的である。ある 1つの問題について、適切な解決を導くために適用可能な戦略は多くあるかもしれない。これは木の OR ノードに帰着する。

一方、suggestion は、元々のゴールを解決するために、1つもしくはそれ以上のサブゴールを導入する。これは木の AND ノードに帰着する。

このような AND/OR への分解によって、最終的なゴールと導入されたサブゴールの関係が追従できるようになった。

・数の感覚 (Feel for Numbers)

back of the envelope 推定を行うにあたって、数の感覚は重要な要素である。ときどき、問題解決の中で正確な数値パラメータを知っている場合がある。また、多くの推定問題では、特定の数や、多い数、少ない数などパラメータに対する値を探索している。

例えば、「良いゲーム用 PC にはいくらかかるか？」といった問題について考えてほしい。RAM やビデオカードは高額だが、それ以外のパーツは普通くらいの値段であるということを知っているかもしれない

このように、あるものに対して、高い、低いといった概念を用いて推論することは、BotE 推論の重要な要素である。そこで、我々はそれらを考慮したシステム CARVE を構築した [Paritosh, 2004]。さらに、量が多い、少ない、といった考えがより正確な推定を導くことを示した。[Paritosh, inpreparation]

Results

BotE-Solver は現在、常識(日常)領域の 13 の問題を解決することができる。システムは、24 の suggestion から成るライブラリを使用している。

ほとんどの場合において、BotE-Solver はおおよその答えを発見することが可能である。基本的には、BotE-Solver のゴールは、推定の際により正確な値に絞り込むことであるが、推定が外れることが興味深い知見を生むこともある。

米国の健康に関する費用を求める問題では、8000 億米ドルと推定した。これは実際の数値のおよそ半分であり、この推定は個人の保険料に基づいたものであった。このようなケースの場合、推定を実行したのちに値の評価を行うということは、プロセスの理解がもっと必要であると分かり、モデルの再考を促すきっかけとなるためである。

5. Conclusions and Future Work

- ・我々は、非常に興味深い概算問題を解決することができるシステム、BotE-Solver を提案した。
- ・また、システムと問題を用いた戦略の知識の分析によって、7つのコア戦略があるということを確認した。
- ・将来の方向として、これらの同じ戦略が問題解決だけでなく、説明活動の中でも使用されることができるかどうかを確認することなども考えられる。例えば、ニュース記事に現われる数の説明を考え出すシステムも考えられるだろう。