人工知能でのプロセスの形　　2014/08/21（木）

自律型人工知能の処理は、ノードの発火の連鎖として基本的に実現できるはずと、ずっと議論してきました。弁証法の実現を機械的な・・・しかも自律的なプログラミングとして行うことを試みます。

１．人工知能の大局的な構造

自律型人工知能は次の３層からなっていると考えるとすっきりします。

（１） 経営層

自律的に、ある期間内に何をなすのか、なすべき作業の作業計画とか、実際に意識的に実行する作業の全体の管理・評価を行います。

（２） 作業目標設定・管理・評価層

各個別プロセスの実行のために、目標設定をし、実行を制御し、結果を評価していきます。

（３） 作業プロセス層

実際の作業を行います。重み付け投票、パターンマッチング、コマンド実行、入力カテゴリー群からある出力カテゴリーへの変換（感情システムとか、重要度評価システムなど）など行います。

作業プロセスは、目標を持たず、とにかく結果を出すことです。実行した結果がどうであるかも感知しません。これは、コンピュータプログラムを実行することに似ています。人間が計画し、コマンドの形に目的の作業を表現し、コンピュータに実行を依頼し、その結果を評価するのも人間です。この人間の作業者の働きを実現するのが、作業目標設定・管理・評価層の作業です。作業は基本的にコマンドです。コマンドは次の形をしています。

（動詞、格、名詞・属性[、格、名詞・属性]・・・）

動詞はどんな作業をするかを表わし、人工知能への入力データや知識データ、現在発火しているノードからの重み付投票で選択されるものです。格や名詞・属性も同様に文脈となる発火したノード群からの重み付投票で決定されるものです。あるいは、コマンドの下位コマンドとして設定されるものかも知れません。重要なことは、ここには人間の関与はないということです。人間の関与はセンサーを通して人工知能に与えられるものだけです。

人工知能全体の行動管理は、規範システムのように作りこまれるべきものがある一方で、機械学習によって、知識として構築されていくものでもあります。知識構築の方向性は作りこみですが、成長は自律的に行われるものです。

２．作業の数式表現

人工知能の機構を考えて行くうえで、重要なことは、プロセスが機械的に実現できることを示すことです。突然降ってわいたような能力で、データができるということが想定されてはいけません。徹頭徹尾機械的作業で行われることを示す必要があります。とするならば、人工知能の全プロセスを数式で表現していくことが良いことであるように思えます。

（１） 重み付投票

S→w1S1,w2S2,・・・wnSn

S,S1,S・・・Sn:ノード。(w1,w2,・・・wn)：投票の重み

ノードSから、S1,S2・・・へ重みw1,w2・・・で投票することを示します。

（２） パターンマッチング

(A) α1,α2,・・・αn→S1,S2,・・・Sm

A:領野。α1,α2・・・マッチングを取るパターン。

領野Aを検索してα1、α2・・・と部分マッチングするノードを見つけることを示します。

（３） 感情システム（重要度評価システムなども含む）

（w1S1,w2S2,・・・,wmSm）→(w1'T1,w2'T2,・・・,wn'Tn)

重み（w1,w2,・・・,wm）でノード（S1,S2,・・・Sm）の発火は重み（w1',w2',・・・wn'）のノード（T1,T2,・・・Tn）への投票を促す。

（４） コマンド

(動詞,格,ノード[,格,ノード]・・・)

という表現で、プロセスを表わします。この下位の処理には、

・コマンド
・プログラム
・マップ解析
・オートマトン
・重み付投票
・パターンマッチング

があります。

（５） プロダクションシステム

パターンマッチングで起動すべきコマンドを選択して行くリストです。

３．弁証法の例

数式表現で弁証法を表わしてみましょう。

（課題）

昔、デモシステムを作ったことがあります。パソコン３台つないで、インターネット環境を再現するというもの。３台だから、画面に３つのパソコンがつないでいるという図を表示するようにしました。ですが、クレーム。本来のシステムは沢山つながれたパソコンの管理をするものだから、沢山のパソコンがつながった図にすべしという。私としては、現実の環境を表示したいと言ったのですが、クレーマーは聞かない。そこで、私は妥協しました。沢山パソコンある図にしますが、３つだけあるエリアに囲って、強調するということに。

（推論の過程）

課題は、次のコマンドを両立するようなコマンドを生成せよというものになります。このコマンド生成自体、いろいろ問題がありますが(パターンを決めて、そのパターン要素へ候補を投票していく・・・という手法になります)、人工知能は、文章からコマンドを生成できたとします。

（コマンド１）実際に３つのパソコンしかないので、パソコンを３つ表示する。

X1:(display,direct_object_case,PC,attribute_case,3,reson_case,A)

A:(be,agent_case,PC,attribute_case,3)

（コマンド２）目的は多数なので、パソコンは多数表示する。

X2:(display,direct_object_case,PC,attribute_case,多数,reson_case,B)

B:(be,agent_case,PC,attribute_case,多数)

(ステップ１)

２つのコマンドの違いを得る。これは、文章が複数あれば自動的に文の統一性の評価として、チェックされるものです。

(find,direct_object_case,”different_case”,between_case,X1,and_case,X2)

このコマンドから、異なる部分が発見されて、(attribute_case,3)対(attribute_case,多数)という結果が得られる。

（ステップ２）

“3”と”多数”の関係を得る。(違いがあることが分かったので、その違いの種類を得るのです)

(find,direct_obuject_case,”relation_case”,between_case,3,and_case,多数)

このコマンドから、”include_case”が発見される。

(include,agent_case,多数,direct_object_case,3)

（ステップ３）
目標を解釈しなおす。違いの種類に応じた、抽象化をします。
（新解釈１）部分を表示したい。
(display,direct_object_case,部分)
（新解釈２）全体を表示したい。
(display,direct_object_case,全体)

（ステップ４）
統合した処理を創生する。（作業目標が、対立する文の両方を満たすことだから、全体と部分の関係をブール代数、もしくはマップ解析で総合イメージを連想していく。そうして、統合イメージからジンテーゼにまとめる）
（ジンテーゼ）「全体と部分を表示したい」と and ロジックでシーンを合成する。
(display,direct_object_case,部分,in_case,全体)

（ステップ５）
ジンテーゼのコマンドから、「部分を全体とは切り離し、強調して表示すべし」と連想する。
「強調」から「色塗りを変える」と連想する。

（ステップ６）最終結果を得る。
（ジンテーゼ）明確に３つのＰＣを多数の中で、強調表示する。
(display,direct_object_case,PC3,in_case,PCmany,attribute_case,"intencity")
(be,agent_case,PC3,object_case,PC,attribute_case,3)
(be,agent_case,PCmany,object_case,PC,attribute_case,多数)

目標設定と結果の評価をしながらコマンドを実行しています。解に近づいていることは、意味が抽象化していって、ジンテーゼを得て、抽象ジンテーゼから具体的コマンドが合成されることを抽象度という基準で評価することで、判断できます。

抽象度は感情システムのビックワード性の評価値によって、重み付投票で決定していきます。

おわり