

データというものは幾つもの塊が合成したもので、データを解析するときには、これらの塊に分離していく作業が必須になっていきます。例えば、日本語文では単語の分かち書き化が必要になります。動画認識では、シーンの区切りを認識するところから始まるでしょう。画像に人が映っていてその人と対話しているとして、まずは、その人を他の画像データから切り取ることが先ず行われるでしょう。そんなデータの塊を認識するのに有効な技術は重み付投票・・・すなわちニューロコンピュータです。手続き的な、データ解析処理、コマンド処理はコンピュータのメイン機能として動作しますが、人工知能では、その解析処理やコマンド処理の節々にデータの塊を切り出すという処理が挟まれるのです。ニューロコンピュータとフォンノイマンコンピュータの密接な相互作用が前提となるということです。

オブジェクト指向で人工知能を実現するとしますと、手続き型プロセスもニューロコンピュータもオブジェクトとして表現されます。それをノードと呼ぶことにしましょう。重み付け投票はパスと呼ぶことにします。すると、ノードをパスにプラグインすることを簡便に行うことが要請されます。オブジェクトのアドレスをもって、ノードへのパスとするのは、論理的でなく、融通の利かないシステムにしてしまいます。ここは、オブジェクトの名前をもって、パスとしましょう。あるいは、名前のハッシュをアドレスと結びつけるから、名前のハッシュをパスにしてもよろしい。とにかく、名前をアドレスに変換する根本機構が想定されて、初めて、論理的にすっきりしたシステムアーキテクチャとなります。

従来、ニューロコンピュータの入力ノードセットと出力ノードセットは人手で固定的に設計し実現されました。しかし、自律的で汎用な人工知能ということだと、あらかじめニューロコンピュータを創りこむことはできません。必要に応じて、グローバルなニューロコンピュータからローカルなニューロコンピュータを切り出して、当面の目標をクリアすべく使えるようにする必要があります。コマンド処理はオブジェクトを創生すればできるように、ローカルニューロコンピュータも簡単に創生できる必要があります。即ち、自己組織化ですね。これができないといけない。

ニューロコンピュータを決めるとき、入力ノードセットと出力ノードセットを設定して、その間をパスで結びます。その時間問題になるのは、いつ創生するかということと、いつ既存ニューロコンピュータを用いてパスを学習して行くかということを決めることです。基本的には、入力ノードセットの名前群と出力ノードセットの名前群が一致していたら、というか、含まれていたら、そのニューロコンピュータを使ってパスを学習し、そうでないとニューロコンピュータを新たに作るということになります。

そのため、名前列の管理が必要になります。基本的に名前をソートしておけば、高速のパターンマッチングができるでしょう。新規のニューロコンピュータを必要とするかどうかはこの名前列管理テーブルを悉皆検索することで実現します。この名前テーブルは、名前列の交差があるかどうかといった、解析にも利用できますから、有効な技術だと思います。

自己組織化は自在に手続き的コマンド処理オブジェクトをつくること、ニューロコンピュータを創ることを実現することです。それらのことは、データの塊というものを認識することが基本です。それはある局面ではフォーカスを定めるものでもあります。フォーカスをどう定めるかもニューロコンピュータの重み付投票で行うことで、そうしたフォーカスを専門に設定するニューロコンピュータ群が存在するということです。メタな機構です。そうして、自己組織化システムは実現できます。

おわり