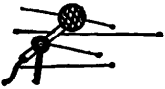


講演

ニューロコンピュータの展望†



甘利 俊一†

はじめに

私の専門は数理工学、つまり数学的な方法論を使って工学的な諸問題に対する一つの切り込み方、方法論を確立する、これを専門としているわけです。この立場から、いろいろなテーマを研究してきました。その中の一つに 20 年ぐらい前から神経情報処理の原理みたいなものに挑戦しました。ですが、やっぱり脳の仕組みというのは難しくよく分からない。たとえば、われわれは頭の中にいろんな記憶をもっていますが、私の頭の中で記憶がいったいどういう形で蓄えられているか、今でも分からない。脳の基本原理はいまだに分かっていないと言っている。ところが、ここ 2、3 年前から、脳でやっているような情報処理を何かテクノロジーとして生かさないかというような話が活性化してきました。

私は 10 年ぐらい前から、情報科学に共通の新しい方法として、情報幾何学というものをつくらにゃいかんと思ひまして、微分幾何を新しく発展させて、情報理論、システム理論、統計的推論に共通の新しい分野と方法を切り拓く努力をしてきました。ところが、数年前からまたニューロコンピュータが盛んになってきて、こちらのほうもやらなければいかんというわけで、大変なわけです。

脳とコンピュータ

現在、脳とコンピュータ、この二つの何が共通で何が違うか、ここにわれわれの関心が集まっているわけです。コンピュータについては、われわれは原理を非常によく知っている。なぜってわれわれがつくったんですから。これに対して、脳だってみんな一つずつもってるんですから、身近なもののはずですが、その仕組みがよく分からない。脳とコンピュータを比べると

当たって、情報処理には実は二つのやり方があったと、ここから話を始めましょう。一つは直列情報処理ないしは逐次型の情報処理。つまりコンピュータが今やっているような情報処理のやり方であり、もう一つは並列の情報処理です。コンピュータは直列型であり、脳は並列の情報処理というわけで、その二つは原理がかなり違っていると言われているわけです。

一言で言えば、逐次直列型の情報処理というのはアルゴリズムを指定して、それに従ってワンステップずつ情報を処理する。それに関する基礎的な理論としてアルゴリズムの理論とか、計算可能性の理論があるわけで、それをバックアップする数学としては数理論理学があるわけです。

それに対して並列情報処理は、非常にたくさんの要素が同時に、いっせいに相互作用をする。何か一つが済んで、次に何が起こってというふうにワンステップずつ進むというよりは、いろんな情報が脳の中に分布して表現されていて、それがいっせいに相互作用する。そうすると、時間とともに相互作用の結果がどんどん進展していきますから、それは一種のダイナミックスになるわけです。もし脳の中に豆電球でもつけておけば、豆電球のあそこが光り、ここが光りとピカピカ非常にきれいなパターンがダイナミックスとともに進行していく。それがわれわれの思考過程みたいなものじゃないかと考えられるわけです。

今まで、あたかもコンピュータは直列の情報処理で人間は並列の情報処理なんだというような言い方でしたけれども、本当はどうかをここでもう一度考え直してみましょう。

情報処理を非常に単純化して分けてしまうと、逐次直列型でワンステップずつ段階を踏んでいくやり方と、何か並列のダイナミックスでいっせいに情報を総合して処理していくやり方がある。人間は実はその両方をやっていると思うんですね。どういうことかと言いますと、ミクロにみれば確かに脳の中でニューロンがいっせいに相互作用をやっているわけです。だけれ

† 昭和 63 年電気・情報関連学会連合大会特別講演(昭和 63 年 10 月 4 日)

場所 新潟大学

†† 東京大学

ど、われわれ人間はアルゴリズムを実行できるわけですね。アルゴリズムを書くこともできるし、アルゴリズムに従って情報処理していくこともできる。人間が論理的に推論していくプロセスは、まさに逐次型に情報を処理しているわけです。

われわれが自分の情報処理の仕方を内観して、俺はこんなふう考えた自分の意識の流れを遡って考えてみれば、まさに逐次的に情報処理をしているんですね。だから人間の中ではこの二つの情報処理のやり方を統一している。両者は階層のレベルが違うんだという言い方もできますし、両方のやり方が同時並行的に相互作用して流れていくんだという言い方もできるわけです。単純にコンピュータは直列で人間は並列なんだと割り切ってしまうわけにはいかない。

逐次情報処理の原理

さて、歴史的に眺めてみますと、人間がどういう情報処理をしているか、情報という観点から人間の仕組みをとらえたいという試みが古くからあった。それを数学的に厳密にとらえたのは、チューリングが1936年に提出した「計算可能性の理論」です。

チューリングは数学者ですから、数理論理学に非常に興味があったわけです。人間が数学をつくり数学の定理を証明するわけですから、当然そういったものに、人間が絡んでくるわけですね。

そのときに人間の情報処理全般をつかもうとしても、全然難しくて無理、いまだに脳の仕組みは分かっていない。チューリングが考えたのが、たとえばアルゴリズム、つまり一定の手順を踏んで情報処理を行う、数学の証明を書き下す、こういうステップを踏んだ論理的な仕方をちゃんとした数学的原理としてつかまえないということだったんだと思うんです。

そのためにチューリングはチューリングマシンというモデルを考案し、アルゴリズムはこれによって全部できるということで、チューリングマシンの万能性を証明したわけですね。

そんなふうにしてチューリングマシンの理論が先にできたんですけれども、チューリングマシンが本当に技術的にこの世の中に実現したのは1946年に電子計算機が生まれたときと言えます。もっとも最初のはプログラム内蔵式じゃなくて、その次のものがプログラム内蔵式だったんですけれども、それはそれとして、1940年代から50年代にかけて情報科学、情報工学が工学の世界にちゃんとした地位を占めるに至る時代

だったと思うんです。

一方1943年にマッカラクとピッツが論文を書く。マッカラクという人は生理学、物理学をやり、ピッツというのは、数学の落ちこぼれと言いますか、高校を出て大学には入らなかった。非常に数理論理学ができる男で、マッカラクがシカゴでその話を聞いて、MITにつれてきた。それで二人で神経、脳のモデルを考えて、ニューロンは一種の論理素子なんだと考えたわけです。ただ、特殊な論理素子、今で言うときい値ロジックという形の論理素子です。それで、ニューロンみたいな素子をもとに、チューリングマシンがつけられることを示した。

情報科学の成立

1940年代に情報工学の設立期としてのいろんな仕事が行われたわけですが、一つはシャノンの情報理論、通信の数学的理論というものを完成しましたし、ウィナーがサイバネティックスを提唱して、動物にしても機械にしても大きなシステムを扱おうとすれば、情報が非常に重要な役割を果たすことを示した。サブシステム間で情報の交換をしなければシステム全体が自立的には働かない。だから通信と制御という問題を扱うことがシステムにとって本質的な問題で、機械系、生物系を問わずにそれを理論的に解明することを言い出すわけです。

そのころ技術としてはコンピュータができました。ここからオートマトンの理論が発展するわけですが、オートマトンの理論そのものはマッカラクとピッツのニューラルシステムに端を発していたわけです。チョムスキーの言語の理論も後から出てくるわけです。

そういうわけで、1940年代から50年代にかけて人は人間のやっているような知的な作業を将来機械的に実現することに夢をはせまして、非常に楽しい時代がきたわけです。すぐにできるかどうかは別にして、電子計算機みたいなものができるんだから、そんなことだって将来できると考えるわけです。

電子計算機は万能チューリングマシンですね。そこで考えてみると、万能チューリングマシンの情報処理のやり方と、人間が脳の中でやっている情報処理のやり方は必ずしも同じとは言えないかもしれない。それぞれに長所短所があって、どっちがいいとか悪いとか言えないけれども、両方ともこれで知的機能を実現するのはまだ非常に大変だという議論になるわけです。

1950年代から60年代にかけて、脳のほうの解明もしたい、コンピュータもどんどん進歩するから、それを使って知的機能を実現させたいという動きがずっとありました。一方心理学も人間の心の働きに迫らなければおもしろいことができないというわけで、認知科学が出てくる。そういうものが一連となって1950年代、60年代非常に盛んだったわけです。

第一次ニューロブーム

きょうはニューロコンピュータの展望をするわけで、後ろを振り返ってみますと、今日のニューロコンピュータのブームの前に、1960年代の初めごろに電気系の人たちがニューロふうの技術にあこがれた時期があったわけです。今から言うとそれは第1次のブームということになるわけです。

その辺の時代といいますと、たとえばイタリアの物理学者のカイヤニエロが脳のモデルというものを提出しまして、大ぼらをふくわけですね。もうちょっと具体的にはローゼンブラットがパーセプトロンというモデルを唱えまして、これは学習する機械です。人間は学習によっていろんなことができるようになるんで、機械だって問題と答えを与えておけば、どんどん、どんどん賢くなって問題を解くことができるんじゃないかというわけです。

パターン認識を考えてみましょう。今では普通のコンピュータでパターン認識の相当複雑なやつができるわけですが、1960年の初めごろをとってみると、パターン認識は非常に難しくて当時のコンピュータでは歯が立たないとされていたわけです。人間は非常に簡単にやっているけど、そのやり方は学習でどんどん、どんどんうまくなっていく。学習する機械の仕組みに人々があこがれたのは当然なわけです。

1960年の前半にパーセプトロン、ほかにウィドローのアダリンとかいろんなモデルが出てくるわけですが、それがはやりまして、電気系の人たちはそういう研究を一生懸命やったものです。

ところが1970年代になりますと、そういう動きがパタッと止まってしまふ。それはなぜでしょうか。アメリカで人々が言っていることは、ミンスキー、例の人工知能の親分のミンスキーとパパートが1968年にパーセプトロンという論文を書いて、パーセプトロンには重大な能力の限界があるといったというんですね。学習するパターン認識装置はあまりうまくいかんということを数学的な理論として研究した。それでみ

んなが腰を抜かしてしまつて、これはあかんわ、こう言いまして、ニューロ関係の研究をみんなやめちゃったと、こうアメリカでは言われているんです。

こうして1960年代の前半非常に盛んだった学習をする機械、神経情報処理ふうの試みはすっかり挫折するとか、下火になる。これは確かなんです。ただそれはミンスキーがそう唱えたからではないと思うんですね。

情報技術の分かれ目

私は技術の歴史からみてそうなる必然性があったんだと思うわけです。1960年ぐらいから、真空管の時代が終わってトランジスタの時代になり、ICの時代になり、人々の想像を超える勢いでコンピュータは進歩してきたんですね。そうするとそれまでパターン認識なんか非常に難しくてできないと言われていたことが、計算力が増えていきますと、腕力主義的にやってもできるようになる。もちろんパターン認識は知的な情報処理というレベルからみれば、低次かもしれない。まだ知的でないかもしれないけれども、でも1940年代、50年代は、いずれ遠い将来の夢物語としていた知的な情報処理が、1960年代の後半に入ってくると、コンピュータの進歩のおかげでとっかかれる時代になってきたのかもしれない、こういうわけですね。

そのときにどういう方法で知的な問題に挑んでいくかなれば、ざっと言うと、逐次直列型と並列型の二つがあるかもしれない。ところで、逐次直列型のほうは原理がはっきり分かっていた。万能性もいわゆる計算可能性の理論で分かっていますし、コンピュータサイエンスの基礎もある。それを実現するハードウェアというものはものすごい勢いで進歩してた。

一方、脳みたいなやり方で実現しようと考えたときに、脳の原理はやっぱりさっぱり分からん。パーセプトロンは単純すぎる。じゃもっと複雑なモデルを作ろうとすると、1960年代のコンピュータではちょっと気のきいたシミュレーションはできないわけです。並列のそういうハードウェアをつくらうかという、当時の技術では絶望的というわけです。

情報処理に直列、並列の二つの道がある。人間は両方を使っているわけですが当時どちらの道を選ぶかと言えば、技術者は当然逐次直列の道を選ぶ。これで人工知能の分野が確立し、多くの人々はこの道にそって人工知能実現の努力をしたわけで、知識工学をはじめとして現在に至る着実な進展がみられるわけです。

それじゃ 70 年代はどうなったのか。工学系の人たちの大勢はニューラルネットワークから去っていったわけですが、生理学の人たちは、微少電極法が 60 年代には非常に盛んになって、本当の脳の中に針を刺して調べていけばもう脳は全部分かる、工学の人のお世話なんか知らずに自分たちで脳のことはみんな分かれる、こう錯覚をしたわけです。そういうわけで、やっぱり工学的なモデルとは離れていってしまう。

1970 年代にはごく少数の人たちが日本でもアメリカでもヨーロッパでも、脳のモデルをもう一度ちゃんと基礎からやり直してみようと考えたわけです。それは、すぐにパターン認識装置をつくるんだとか、人工知能を実現するんだとか、そういうことはもう夢見ないで、神経回路網に特徴的な原理は何なのか、その可能性は何で限界はどこにあるのか、そういうことを詰めていこうではないかという地味な基礎研究の時代が 70 年代を通じて 10 年間あった。

1980 年代に入って話がちょっと変わってきたわけです。研究者はこれまでに 20 数年間人工知能を研究してきたわけです。人工知能もいろんな紆余曲折を経て、数学の定理を証明するんだとか、ゲームを解くんだとかいうような話から、もうちょっと現実的なものとしてエキスパートシステムをやるんだというようになってきたわけです。で、1980 年代の初めに人工知能ブームを迎えて、ついに人工知能が実用になるようなことが宣伝されたわけですが、やっぱりそうはなかなか問屋が卸さない。

やっぱり逐次直列型の情報処理一本やりですべての情報を解き明かすのは無理なんではないかという反省が生まれてきた。人間はどうも直列と並列の両方を非常にうまく使ってやっているらしい。1960 年代の終わりに技術化の見込みがないからほっておけといった脳について、そろそろちゃんと研究したほうがいいんじゃないかという技術的な関心がまた 1980 年代に復活してくるわけです。

人工知能の限界

一番単純なエキスパートシステムを考えてみましょう。たとえば、医者診断システムです。これを作るには、お医者さんのところに行って医者の知識と医者の推論の仕方を全部聞き出してくる。それをもとに知識データベースをつくり、if-then, ルールをつかって、何かうまく推論をやれば、並の医者よりはましなシステムがつくれる、こういう話があるわけです。

そこで、医者のところに行きまして「先生、あなたは患者をどういうふう診断しますか」と聞き出して、病気の診断の知識をコンピュータに全部ぶち込む。医者がどういう推論規則を用いて患者のデータをみて推論するかを調べるために、名医に「あなたはこの患者をみて病気は何だと判断したけれども、それはどういうわけですか」と聞くわけです。医者が「わたしは名医だからこうこういう理由で、こういう推論を下すんだ」と言うと、それを全部規則として書き並べまして if-then ルールをつくる。

そうやってつくった医療診断システムは医者と同じことをやるかという、それは全然違うと思うわけです。医者はどうやってるかといいますと、たぶん患者をみて、いろんな診断のルーチンを行う。そのうちにいろんなデータが集まってくると、if-then ルールを一つずつあてはめて推論するんじゃないで、「うーむ」とにらんでるうちに、もやもやと答えが突如として出てくる。この患者はきっとこういう病気ではなかろうかと、いうわけです。どうやってそれを思いつかかそのプロセスを医者は論理的に説明できません。だから、それは直観であるという。「わたしは非常にカンがいいんじゃない」こう言うわけです。

さて、ヤマカンで答えを出したあとで、ヤマカンがそのまま本当だとは医者も思っちゃいない。そこで自分の知識と経験と、それから推論規則を当てはめて、自分のヤマカンが本当に正しいかどうかをチェックするわけです。AI の専門家が来て医者に「あなたの診断のプロセスを説明してください」、医者が「よしよし、わしの経験を教えてやろう」、こういうときの説明は論理的にチェックする部分の話なんです。それならちゃんとした論理的な推論です。

論理的なプロセスと直観的なプロセスとか交互に進んでいくんだけれども、直観のほうは人に説明できませんから、AI の専門家が「あなたはどうやって診断したんですか」と言えば、「わたしはこうだ」と理路整然と論理的プロセスを説明するわけですね。だけれども人間の頭がうまく働くのは、直観の部分と、逐次直列型の論理の部分とが非常にうまくみ合っているからなのではなかろうか。そうだとすれば、やっぱり今まで逐次記号処理による論理の道をずっと理論化してきたけれども、こちらの並列処理の道もそろそろ考えてみる段階にきているのではなかろうか。これが 1980 年代の風潮になりつつあるんですね。

ハードウェア技術のサポート

それにはやっぱりテクノロジー進歩が見逃せない。ご承知のとおりコンピュータ技術、LSIの技術がものすごい勢いで進歩してきたわけです。今ならばマイクロプロセッサを1万台とか、極端に言えば100万台並べた高密度並列装置をつくらうと思えばつくれる。LSIのニューロチップだって、つくれと言われれば技術的には実現する可能性がある。さらに高密度に結線するのが難しいとなれば、光技術を使ってやるのが可能かもしれない。1960年代とは技術的に全然違った時代を迎えたわけです。

ところが、100万台のプロセッサを全部つないで計算をさせたときに効率が上がるかと言えばあんまり上がらない。一つの問題を100万台に分解してやるとすれば、お互いのコミュニケーションの時間ばかりかかっちゃってかえって調子が悪いわけです。普通の逐次型のアルゴリズムを並列のシステムで計算させようと思えば、100台ぐらいの並列ならば効果が上がるかもしれない。だけれども100万台の並列じゃかえって効果が落ちるにきまっている。今までのプログラムとは全然違った何か、いわゆる脳の中でやっているような相互作用の情報処理、これを取り入れれば超並列の効果が上がるかもしれない。

それならば、脳でやっている計算とは何か、今この技術をちゃんと考えるべき時期にきたのではないかというわけです。今から研究を開始しておけば、2000年代に入ってテクノロジーとして開花するんじゃないか、こういう研究開始の時代を迎えているんじゃないだろうか、その上で直列・並列の二つの技術を融合したものができなきゃいけないんじゃないか。これがニューロコンピュータの展望ということになるわけです。

ニューロンの動作

ごく簡単に神経細胞のモデルをお話ししておかないと、話が進みません。ニューロンはそこに入ってくる多数の入力、 $x_1, \dots, x_n$ を総合して計算して自分の答えを出す。こういうエレメントです。

このニューロンが取り扱う入出力情報 $x_1, \dots, x_n$ はアナログ情報だろうと考えられています。具体的にはニューロンが保持している情報はパルス頻度で符号化されたアナログ情報です。つまり1個1個のニューロンがファイアするか、しないかという非常にミクロな時間スケールでみないで、ある程度の長い時間スケ-

ールでみましょう。ここにコップがあったたと、喉が乾いたらこの水飲めばうまいかな、あったまっちゃってあんまりおいしそうじゃあないかなと、そんなことを私が考えるときに、それが脳の中でどういう形で表現されているかと言えば、いろんなパルス頻度でいろいろなニューロンがパツ、パツ、パツと興奮してるに違いないんですね。

水がどういう形で私の脳の中で表現されているかわかりませんが、それは単純なパルスというよりはいろんな場所に分布したニューロンが、いろんな時間頻度でパツとパルスをを出していく、そういう形で分散表現されていると考えるわけです。

ごく簡単に言いますと、ニューロンは入力情報の荷重加算をしています。入力情報 x_1 は重要だから s_1 倍しよう、この情報 x_2 は s_2 倍しよう、この x_n は s_n 倍しようというふうに、それぞれの入力情報に対応する倍率が決まっています。それを掛け算してあとは全部を総和する。総和が定値 h を超えないと答えは出ず出力はゼロになる。ゼロというのはパルスを1個も出さないということなんです。入力の重みつき総和が h を超えますと、その程度に応じた頻度で出力パルスがパ、パ、パと出る。だけれども、総和に応じて非線形関数を使って出力にするのはめんどくさいということになれば、総和が小さければ答えを出さなくて出力はゼロ、総和が大きければ、バアツと最高頻度で興奮し出力は1であるということで、入出力信号はゼロか1で近似する場合も多くある。各入力信号の強さに重みをつけて、つまり効率をかけて足し算すると言いましたが、この重みつまり効率にはプラスのものとマイナスのものがある。これがかなり重要になってくるんです。

x_1 にかかる重み s_1 がプラスで非常に大きければ、 x_1 に信号がくると、それは s_1 倍されて非常に大きくなりますから、この信号はこのニューロンを興奮させて強い答えを出させるのに非常に効果がある。仮に s_1 がマイナスだとすれば、 x_1 に信号がくると、 $s_1 x_1$ はマイナスで引き算になっちゃうわけです。ほかの入力が興奮させようと思って信号を送り込んでも、この倍率がマイナスでものすごく大きいとなると、興奮を抑さえちゃうわけです。

こうして、二つのニューロンはその重みがプラスでつながっているか、マイナスでつながっているかで、異なる相互作用をする。

二つのニューロンの間の相互作用は、興奮性でつな

がって相手を興奮させようとする興奮性のものと、マイナスでつながっていて、相手の興奮を抑えてしまうという抑制性のものとがあり、これがうまくバランスして、全体でネットワーク組んでいる。

それから、学習とか、記憶を議論しなければいけないわけで、学習というのはいったいどういうふうに行進していくんだと言いますと、ニューロンとニューロンのつながり方の重みつまり、効率が変わっていくんだとしか考えられない。これもずいぶん昔から言われてたんですけど、最近の生理学ではニューロンの結合の効率がよく変わる、早いものでは数秒でつなぎ方の効率が変わることが最近分かってきたわけです。そういうわけで、ニューロンにとってみれば、ある信号 x_1 を信用して結合の重み s_1 を非常に強くしていたのが、どうも自分の動作がうまくいかん。よく考えてみると、信号 x_1 を信用しすぎたからいかんのだということが分かれば、 s_1 をどんどん切り下げればいいわけですね。もっとましな信号が x_2 かもしれない。そっちがいいんだとなれば、 s_2 をどんどん強めるわけです。そんなことがネットワーク全体の中で自動的に行われて、その結果学習が進むと考えられているわけです。

ニューラルネットの動作

さて、ニューロン回路の数理的なモデルを地道につくってみましょう。たとえば、入力情報 $x_1 \cdots x_n$ が回路網に共通に入ってきます。そうしますと、各ニューロンはこの入力信号を全部集めて計算して自分の答えを出す。どのニューロンも同じ情報を受け取って計算して答えを出すとしても、答えは同じかというと、そうじゃなくて、各ニューロンはどの信号を重視しどれを重視しないか、つまり結合の重みがニューロンごとに違う。だから同じ情報を受け取っても違う答えになる。こうみると、この回路網は、入力信号 $x = (x_1 \cdots x_n)$ を受け取って出力情報 $z = (z_1, \cdots, z_m)$ に変換する変換装置ということになる。

ニューロンネットワークの基本的な動作は入力信号のベクトル空間 X から出力信号のつく空間 Z への情報変換である。ニューロンのような素子を用いて行う情報変換で一番特徴的なものは何なのか、これをつかまえないかいけないわけです。

情報空間の変換構造は、もちろん結合がなんであるかによって大きく違うんですけども、任意の変換が実現できるわけではない。こういうものに特徴的な変

換というものは何かを数学的にきっちりとした形で出そうというのが、われわれ数理工学という立場からの議論なんです。

ところで、脳の中ではニューロン同士が相互につながってフィードバックのあるもう少し複雑な回路を考えなくてはならない。その場合はおのおののニューロンが計算して出した答えがまたフィードバックして自分のところの入力へ戻ってくる。いろいろのニューロンの答えがほかのニューロンに入力する。

こういう神経回路網の動作がどういうふうになるかと言いますと、時刻 t の回路の状態 x_t として、あるニューロンが興奮したりしなかったりという、そういう興奮の分布を考える。次の時間にはこの x_t が入力情報となって、各ニューロンはこの情報をもとに入力の重みつき総和を自分で計算をして、自分がどのくらいの強さで興奮したらいいかを定めるわけです。そういうことをニューロン全部がやるわけです。

ニューロンの現在の状態を状態空間に書きますと、現在ここにいればそれをもとに計算して、次は状態がここにくるというふうに、状態がどう遷移するかを示す状態遷移図が書けるわけです。状態遷移図は、各状態からその次の状態へ矢印を引いたものです。その中には、次の状態が実は変わらないで自分のままという平衡状態がありまして、いっぺんこの平衡状態に入っちゃうと、あとはいつまでもここに止まっているということになります。また、今状態がここにあると、ここに行き、ここに行き、また元に戻るというぐるぐる回りを続ける周期状態もある。

もちろん外から新しい入力情報が入ってきますと、状態遷移図はまた変わっちゃいます。こういうニューラルネットワークに特徴的な状態遷移の様相というのは何なのか、これを調べる必要がある。その上で、情報処理のダイナミックスがこういうネットワークのダイナミックス上に実現できるかを考えたい。これが並列情報処理の一つの基本だというわけです。

逐次型の問題解決

並列のダイナミックスにおける情報処理というのはどういうものかというのを、素人分かりのする例でちょっと示してみましょう。

有名な例ですが、「Time flies like an arrow.」という文がある。この文章の意味を確定しましょう。「光陰矢の如し」というわけですね。

これがなんで問題かと言えば、time という単語を

字引を引くと、「時」という名詞と「タイミングをとる」という動詞の二つがあって、これだけでは、どちらがよく分からない。fly にも「ハエ」と「飛ぶ」がある。like にも、「好き」と「～のように」がある。

このときどうして「光陰矢の如し」というふうに全体を解釈するのかと言えば、文章全体を総括して文法的にも意味的にも筋の通った解釈を人間は求めるんだと、こういうわけです。

いま、逐次直列型の情報処理でこの解釈プロセスを確定するコンピュータプログラムをつくってみたいとしましょう。逐次直列型で考えてみれば、time に「時」と「タイミングを取る」という2通りの意味があれば、可能性が二つに分かれる。fly に「ハエ」と「飛ぶ」という2通りの可能性があれば、それがまた前のおおのの可能性に対してさらに二つに分かれる。さらに like に2通りの可能性があれば、それぞれが2通りに分かれるというふうに、可能な解釈のどちらを取るかで2通りずつに分かれたツリーがつくれる。

ここで変な組合せを選んで、たとえば time も動詞にして、fly も動詞にすると、その組合せは文法的に矛盾する。またある組合せを取ると意味的にナンセンスになってどうも調子が悪いとか、そういうことが出てきます。ツリー上で駄目な組合せのところにバツが書いてあるんですね。そういう組合せは規則に違反するということになるわけです。

こういう状況のもとで規則に違反しない組合せを求めてツリーをサーチをしていく。どうやったら一番効率よくサーチができるかということで、デプスファーストだとか、ブレッドファーストだとか、いろんなサーチがあるわけです。

こういうサーチを1個ずつやっていくのは厳密でいいんだけど、一般の問題をそういうふうにツリー上で表現して可能な組合せの中から一番いいやつを一つ選ぶというのは、一般的に言えば1個1個の要素の選び方に2通りずつあって、要素が N 個あれば、2の N 乗の可能性の中から一つ選ばなければならない。これが組合せ論的爆発と言われているものになるわけです。

それからもう一つ、こういうプロセスでやってるときに、バツが書いてあって、これは文法的におかしいとか、これは意味的におかしいというときは、そこでもう探索を打ち切って別の可能性を探すんだということになりました。けれども、なんとなくおかしいけれど、ひょっとするといいのかもしれない。それはもっ

とほかの要素を調べたいという、いいか悪いかははっきりしない組合せもあるだろう。つまりもっとも多数の要素を総合しておかしいかどうかを決めなければいけないときに、こういうやり方はちょっと困るんですね。順々に決めていって、ここはこれおかしい、ではバックトラックして前に戻るといようなことならよいが、なんとなくじわじわとおかしいので、もうちょっと後までみたいというプロセスがうまく表現できないわけですね。

ニューラルネットによる問題解決

さて、こういうような文章解釈みたいな論理的問題を、並列のダイナミクスを使って情報処理をする可能性があるんだろうかということを考えてみましょう。そうすると可能性はあるんですね。今のテクノロジーのもとですぐに現実性があるのかどうか、これまた別問題。

それはどうするのか、並列のネットワークでどうするのかを考えてみましょう。

time に「時」と「タイミングを取る」という二つの解釈があるならば、それぞれの解釈に対応して二つのノードを置きましょう。解釈ノードが一つのニューロンであるかどうかはあやしいですが、分かりやすくするためにここではニューロンと言っておきます。1個のニューロンがそんな高級なことをやっているとも思えませんけれども、たとえばの話です。この二つを「時ニューロン」と「タイミングニューロン」と呼びます。fly に2通りの解釈があるならば、ニューロンを二つ付ける。「ハエニューロン」と「飛ぶニューロン」としましょうと、like についても「好きだニューロン」と「のようにニューロン」。こうした解釈ニューロンを多数並べて、それらが全部相互に結合してネットワークをつくるというわけです。

それで文章をバーッと見て「ウン、光陰矢の如し」だなとわれわれが思ったとしますと、これが並列の情報処理ではどういうふうに表現されるのか、time という単語だけを取れば、時ニューロンも、タイミングニューロンもどちらも興奮する可能性がある。しかし全体で解釈が決まるときは、時ニューロンだけが興奮し、ハエニューロンは黙っていて、飛ぶニューロンだけが興奮し、という具合で取り上げた可能性の方だけが興奮している。それを見ると、ああなるほど光陰矢の如しだなという解釈ができる。これがネットワーク上での情報表現です。

私の脳の中で本当にそういうふうになっているんだと言ってるわけじゃないです。たとえ話で言っているわけでありませう。

さて、こういうネットワークで正しい解釈がどうして作り出せるのか、その可能性について議論してみましょう。たとえば fly を取りますと、これに対応してハエニューロンと飛ぶニューロンがあったときに、両方が興奮したんじゃない困るんで、どちらか片方が生き残って興奮する、こういう状況が実現しなければいけないわけです。それを実現させるためには、一方のニューロンが興奮するとその答えが他方のニューロンに非常に強いマイナスの結合で入ってくればよい。一方が興奮すれば他方の興奮をやめさせるくらい強力なマイナスの結合が入ってくることになります。一つの単語に2通りの解釈があれば、両雄並び立たずでどちらかが勝たなきゃいけませんから、両方はけんかをしてどっちか勝ったほうが相手を押さえつけるんだという非常に強いマイナスで結合している。

一方、二つの単語で、ある解釈とある解釈とは非常に相性がいいとする。「時が飛ぶ」というのは相性がいいかどうか知りませんが、名詞と動詞というのは二つ組むのは相性がいいんだとすれば、そういう相性のいい解釈はお互いにプラスで結んでおいて、一方が興奮すると他方の解釈も興奮しやすくしておく。

今度は動詞が二つ続くというような解釈はおかしいとなれば、その二つはマイナスでつなぐ格好で、ネットワークを作る。

こんなふうに、結合をずらっと張りめぐらしたらどういことが起こるだろう。そうしますと、与えられた文章をパッと読んだときに、初めに何が起こるかという、たとえば fly だけをパッと見れば、ハエの可能性もあれば、飛ぶという可能性もありますから、両方のニューロンがとりあえず弱々しく興奮する。その単語だけで解釈が決まるわけじゃないから、それは弱々しくしか興奮しない。time についても、like についても二つの解釈が弱々しく興奮する。

そのうち何が起こるかという、いくつかの単語の解釈で、お互いに全部プラスにつながっているような非常に相性のいい部分的な解釈が見つかる。それらはお互いに助け合ってどんどん、どんどん興奮すると、「お前もっと興奮しろ」と他に援軍を送りますから、グループ全体がお互いに馴れ合いでどんどん、どんどん興奮が強くなるわけです。

こうしてローカルに興奮する部分グループが方々に

で上がらなければならない、その間でお互いにマイナスでつながっている部分があると、そこが今度はけんかを始めるわけですね。そうやって何かダイナミックなプロセスが進行した結果、最後にネットワークの安定の興奮状態として出てくるのは、何か意味の通った解釈ではなからうか、こう言うわけですね。もちろん、これでうまくいくかどうか、数学的に調べる必要があります。ニューラルネットワークがあって、相互にプラスやマイナスでつながり、それがダイナミックスをした結果、ちゃんと正しい解釈へ行きますか、それとも周期解なんかに入って、ぐるぐる回りになって、解釈がいつも安定しないと困る。うまいところに本当に収束して、そこがちゃんと「光陰矢の如し」という解釈になってますかというわけです。

ニューロダイナミックスの問題点

本当の脳の話はしばらく忘れて、テクノロジーの話として並列の情報処理を考えるということになれば、ニューロふうの回路を考えて、たとえば文章の組合せ論的な解釈が、こういうネットワークの上に表現できるという問題になってくる。もちろんどっか初期状態からスタートしてダイナミックスの収束の時間が、2の N 乗なんていうように長いんだと、これは困るわけですね。だからこれはどのくらいの長さなのか、数学的な問題として一般にこういうネットワークで典型的な動作の収束時間はどのくらいなのか、きちんと調べなければいけませんね。

たとえば、あるニューロンからあるニューロンへの結合が逆のニューロンから元のニューロンへの結合といつも同じという、本当の脳にはちょっとありそうにもない条件を満たしていれば、周期解はなく、回路のダイナミックスはどこか安定状態に必ず収束することが分かっているわけです。

それで、今の場合に本当にうまくいくかといいますとなかなか難しいわけでありまして、相互作用の結合を、これとこれとは相性がいいとか、これとこれとはちょっと矛盾するからマイナスにつないでおこうとか、うまい結合をつくったとして、そういうネットワークはどうなるかといいますと、実は平衡状態がたくさんできる可能性がある。一番いい答えが一つだけ安定になるというふうにはいかない。いったいどういう状況にすればちゃんと望ましい一番いい答えが一つだけ出るかということが、非常に大きな問題として浮かんてくる。

ただ、一般的に言えることは、この例の場合には、「時は矢のように飛んでいく」という解釈だと思えば、そうも思えるけれども、この文章はそんな高級なことを言っているというもんじゃないとも思えるわけであり、タイムフライというのは、時バエという新種のハエでありまして、アマゾンか何かで最近発見されたんだという噂はあまり聞きませんが、そう言われれば、ああそうかなと思うでしょう。この文章は「時バエ矢を好む」(笑)。矢みたいなのに時バエが止まっていて、蚊かなんかが飛んでくるのを待っている。蚊を常食にしている、矢の上に止まっている新種のハエである、と解釈しますと、これももちろん文法的にも意味的にも安定した解釈になる。この例は、二つの解釈のどちらもが安定に成り立つ例でありますけれども、ニューロンのダイナミックスは一般には平衡状態が一つじゃなくたくさんあっていいんじゃないか、こういうわけです。人間だって、正解がいつも一つとは限らない。

現在ニューロンふうの回路で問題を解くいろんな試みがされているわけです。巡回セールスマン問題とかいろんな組合せ問題があるんですが、そういう組合せ問題を今言ったニューロンのネットワークの並列のダイナミックスの中に埋め込んでしまふ。初期状態からニューロン間の相互作用をさせた結果いい答えが出てくればよい。その答えは非常に早い時間で出てくるんではなからうか。

一般にニューラルネットワークの相互作用の結果、状態が収束する時間というのは非常に早いと言われてます。ニューロンの数が N だとすれば、ある条件のもとではだいたい $\log N$ ステップぐらいで収束するんじゃないか。これ非常に早いですね。2の N 乗というのは、これは組合せ的爆発で困る。2の N 乗の可能性から一つを選ぶのに $\log N$ というオーダーですから、それは非常にいい。しかし一方では、下手につくると、いろいろ平衡状態が方々にできてしまふ。今のように「時バエ矢を好む」と「光陰矢の如し」と二つが平衡状態で、あとはないというんだと、これは大変都合がいいんですけども、まだまだ変な解釈がいっぱい出てきちゃうかもしれません。こんなことが起こるんで、問題をうまく解決するような現実の並列計算法がどうすればできるか、これが問題です。

並列処理と近似解

ホップフィールドという理論物理学者がいて、アメ

リカではなんとなくニューロンネットワークの元祖、神様にされていますが、巡回セールスマン問題のような組合せ問題をニューラルネットワークで解いてみせた。ダイナミックスに埋め込んでみて、理論的にうまくいくかどうか分からんが試しにシミュレーションでやってみると、割合うまくいく。どううまくいくのかというと、ニューラルネットワークはやっぱり安定平衡状態がいっぱいできるから、初期条件を何にするかでいつも違った答えを出しちゃうんですね。ところが、ホップフィールドが言うには、並列情報処理をする場合には組合せ論的にみて本当に一番いい答えを探そうというのは、これは非常に難しいだろうと。それには本当に論理的にステップを踏んで解いていかなければいけないけれども、論理的にステップを踏んで全部解いていくということは、組合せ論的爆発を覚悟しなればできない。巡回セールスマン問題はNPコンプリートの問題で、要するに非常に手間がかかる問題だというわけです。

ところが、並列のダイナミックスでやれば、いろんな条件を総合的にみた比較的好い答えが出る。一番いい答えじゃないかもしれないけれども、十分満足に近い答えを出すではないか。人間もそうだし、動物でも運動制御をし、いろいろ日常のデシジョンを下すのは、ただ論理的な組合せを全部調べて最適な解を出しているとはとうてい思えない。非常に早い、瞬間的に答えを出さなければいけないような場合には、最適だとは言わないまでも、かなりいい解が素早く出ればいい、こういうわけです。それは一番初めに例にあげたように、お医者さんが直観でいろんな条件を総合して、無意識のプロセスでニューロンネットワークのダイナミックスを繰り返して、答えを出すというのと同じようなことを並列のネットワークはいつもヤマカンでやっていて、人間の場合にはそれを意識にのせてもう一度ちゃんとステップを踏んで論理的にチェックしようという機能が働いてるわけですね。

動物の場合にはチェックしようという機能がどこまで働いているかよく分かりませんが、失敗すればやっぱり学習が働いてダイナミックスをよくする。そういうことで並列計算の可能性、つまりどこに可能性があり、どこに限界があるかということをもうちょっとちゃんと確かめてみる必要があるんじゃないかというわけです。

そのときに、直列の逐次型の計算のように、最適解を必ず出すんだという立場には立たないで、それとは

全然違った視点から新しい計算の理論をつくっていかねばいけない。計算複雑性の理論も、今度は並列回路の複雑さがからんでくる。それが来るべき並列計算の可能性の理論になるのではなからうか。

それについては、ボルツマンマシンといいますが、計算のステップに確率的な性格を導入することによって、ダイナミックスの過程で非常にうまいところへ収束させるようなアイデアもあります。

ニューラルネットの学習

学習に関してもいろいろな話があります。ここでまた昔のパーセプトロンに戻りますけれども、入力情報があって、それを一種のランダムに結合した回路網で情報変換を行っておいて、その答えを総合して最終層のニューロンが答えを出す。この答えによってパターン認識を行うというのが1960年代にはやったパーセプトロンの考え方ですね。

分かりやすく言えば、たとえば男と女を見分ける機械をつくらうというわけです。男の顔がきたら最終層の出力ニューロンが興奮して男だということになる。女の顔がきたら興奮しないで、今は女だよという答えになる。もちろんもう一つ女ニューロンというのをつくっておいて、女がきたらこれが興奮して男がきたら興奮しない、これでもよい。そういう装置を学習でつくらうということを初期のパーセプトロンは考えたわけです。

パーセプトロンは、男がきたのに女だという答えを出したら、先生が「こら、このニューロン、お前は間違えたんだ」と言うので、この最終層の出力ニューロンが「はい、すみません」と言って自分が受け取る入力信号の結合の重みを変えていく。そうすることでパターン認識が正しく学習できるのではないかというのが、初期のパーセプトロンの考えだったわけです。

本当にうまくいくのかというと、うまくいく場合といかない場合がある。どういう場合にうまくいかないかも分かった。そのときに、入力信号を変換する中間素子のニューロンの数をもっと増やして、 N 個の入力情報に対して2の N 乗ぐらいにすれば、これは万能性があることが分かったわけです。だけど2の N 乗個も使ったのでは困る。組合せを全部調べることになるから。そうしないですぐうまくいくにはどうしたらいいか。

そこで考えられるのは、中間の変換ニューロンが何を計算するかその役割を固定してしまわないで、これ

からのニューロンも学習するようにすればいいじゃないかというわけです。そうすれば全体の能力がずっと上がるのではないか。こういうわけです。

そういう学習の仕組みがうまくいくかというと、ちょっと問題がある。最後の出力ニューロンが間違えると先生が来て「お前がドジ踏んだ」というので、このニューロンは結合を少し変えようじゃないかとなる。しかしこのままでは何べんやってもうまくいかないかもしれない。そうすると、この出力ニューロンが考えるわけです。どう工夫してもうまくいかないのは、前段階の中間ニューロンが必要な情報を計算して自分にくれないからだ。これが文句を言うわけです。先生おればかり怒ってもだめだから、こっちの中間ニューロンを怒ってください。

そこで先生が「なるほどこっちがアホなんだ」と言うんだけど、中間ニューロンのどれが間違っているかを判定するのは非常に問題があるわけです。つまり特定の1個が何をするかというよりは、全体で組み合わせて答えを出すのに必要な情報を分担して計算するので、一つ一つをみてもどれが悪いかわからない。

これは学習の根本なんです。結果が悪い。動作が悪かったというのはやってみれば分かるから、今出力ニューロンが計算した答えがまずかったならすぐ分かるけれども、このずっと前のほうの途中の段階でどのニューロンが悪くて、どのニューロンがよかったかというようなことを分析する仕掛けは脳は自分でつくれないんじゃないかならうか。どれが悪かったかは、やっぱりニューロン同士がお互いに話し合って自動的に決めなければいけないので、外部からの先生はそこまでは教えてくれない。それでどう話し合って学習をすればいいかというような議論がいろいろ進んでいるわけです。

バックプロパゲーション学習

ご承知のとおり、バックプロパゲーションという学習方法が提案されて、今これではやっているんです。それは別に難しいやり方ではなくて、出力ニューロンに誤りが起こると、ここはすみませんと言って自分の結合を修正するんだけど、悪いのは自分だけではないと言って、他にも責任を分担してもらおうべく、誤り信号を逆方向に送り込んでやるわけです。それによって他のニューロンも少しずつ勉強していけば、何かうまくことが起こるのではなからうか、こういうわけです。

バックプロパゲーションを提案した人たちは、ミニスキーはパーセプトロンの限界を議論したんだけど、中間層ニューロンが学習するという可能性までは考えていなかった。それを見つけたから、新しい突破口が開けたと主張したんですが、そんなことはないんでして、中間層をどう学習すればいいかという話は昔からあった。

自分のことで恐縮ですけども、1968年に共立の情報科学講座の1冊に中間層ニューロンの学習の話を書きまして、IEEEのコンピュータ誌にも書いたんですけども、学習方法はちゃんとあることが分かっていた。では、当時と今と何が違うかというと、今ならいろんな例題で、学習してみたらどのくらい強力なパターン認識系をつくれるかをコンピュータシミュレーションでできる。当時はできなかった。今の人たちはいろんな例でやってみたら、思ったよりうまくいくではないかと言ってます。これで、再発見されたこの方法がブームになった。いま非常にはやっております。最近ニューロコンピュータという、バックプロパゲーションとなっている感がある。しかし、これにもやっぱりいろいろ問題はあるわけでありまして、よく言われるのはスケール効果というやつで、小規模の問題ならばこれで思ったよりはずっとうまくいくけれども、本格的な問題をやろうとすれば、やっぱりうまくいかない。そのためにはニューラルネットワークのアーキテクチャがまだ単純すぎる。もっともっと本格的なものを考えなければいけないんだけど、そうするとこんな単純な学習法、今提案されているような簡単な学習法というのは、実はローカルミニマムに落ち込んでしまう可能性がどんどん強くなってうまくいかないのみか、学習時間が非常に長くなるなど、いろんな欠点が出てくるわけです。

学習の分野に関しても、ニューロコンピュータの可能性を探すとすれば、これから時間をかけてその可能性を原理のうえで探していかなければいけない。こういう地点に立っている。

例題からの学習

ニューラルネットワークの学習による情報処理と、たとえば人工知能ふうにつくったパターン認識装置といった何が違うのかということをもう一度整理してみよう。今パターン認識として男の顔と女の顔を見分けるために、どういうコンピュータプログラムをつくらなければいけないかと考えますと、まずわれわ

れは男の顔の本質的な特徴は何なのか、女の顔というのは本質的には何なのかということを知識として知っていないことにはつくれないわけです。

これは難しいですね。人間だって写真だけでは男か女かなかなか見分けられっこない。ですがわれわれが9割くらい当てられるとすれば、なんで当てているんだろうか。目の長さで当てているのか、鼻の格好で当てているのか、口だけで当てているのか、全体を用いているのか、よく分からない。人工知能ふうの設計は、設計者が全部知識を知っていて、それをコンピュータに入力する。それに対して、学習の立場でやれば、例題だけどんどん、どんどんやっておけばいいんだというわけですね。

学習した結果、後でネットワーク中のニューロンが何を計算しているかを調べてみれば、なるほど男と女を見分けるにはこういう特徴が重要だったのかということがちゃんと分かるではないかというわけです。そういう意味では特徴発見のヒューリスティックスとしてもこういうモデルは使えるという議論があるわけです。

話は長くなるからやめますけれども、外界の情報が脳内で表現されているとすれば、それは分布した情報の形で表現されているのか、それとも何か局在化した情報で表現されているのかという議論があります。脳はたぶん両方の仕組みをうまく使って、あるときは情報を分散させダイナミックスの相互作用を起こりやすいようにし、あるときは情報を局在化しシンボライズして固定する。そのシンボルを使って逐次型の情報処理へ移行できるようにしておくわけです。そういう情報表現が起こる神経機構の数理的な可能性を議論しなければいけない。

並列情報処理の原理

さて、ニューロコンピュータの展望をしてみましょ。直列逐次型の情報の原理というのはアルゴリズムの理論、計算可能性の理論はわれわれはすでに知っているわけで、その上にコンピュータサイエンスの体系が築かれているわけですね。それと同じことで、並列情報処理の可能性、万能性と限界を原理の問題としてちゃんとつかまえておかなければいかん。

これはたぶんアーキテクチャに依存していて、どういうアーキテクチャのニューラルネットワークをもってくれば、どういう点が強力でどういう点に限界があるということがいろいろ出てくるだろう。本当の脳の

回路がどうなっているかはちょっと別問題として、いろいろな回路の原理を一つの数理的な構造としてわれわれは研究していく必要がある。その立場からついに本当の脳を調べてみれば、本当の脳はこういう情報原理を生物学的にいろんな制約のうえで実現していることが分かるだろうというわけです。

われわれの頭がこんなにうまく働いているということは、並列情報原理は結構うまくいくんだという存在証明を与える。一方この情報原理を、脳とは別に工学的な立場で設計してこれを実現すればニューロコンピュータになるというわけです。

そのためには、ニューロコンピュータを実現するための並列のハードウェアのテクノロジー、LSI でも光コンピュータでもいいんですけども、こういうものを研究していく必要がある。

正直言いまして、われわれはニューロコンピュータなんてうさんくさい名前を使っていますけれども、並列のマシンで学習能力をもつ情報処理の原理が技術的に実用化されるのは、それはずっと先のことでありまして、21 世紀に花栄える技術であると思っているわけです。

もちろん音声情報処理だとか、画像情報処理だとか、比較的低次のやつで、しかもかなり情報の次元が多くてアナログ色の強いものは、ニューロと言おうが言うまいが、少しずつ並列性の処理が実現していく。これはニューロと言わなくたって、今までもそういうアナログ的な情報技術はパターン認識にあったわけです。その連続的な延長上にニューロふうも加味した技術が実現していくと思うんです。

新しい情報技術の体系

もっと将来のことを考えれば、並列情報原理を軸とした、情報科学の新しい体系というものが当然できてしかるべきなんで、それは新しい脳の科学といってよい。生物学としての脳の科学と、情報原理からみた並列情報処理の科学がつながった大きな一つの学問分野をこれから形成していくのではなからうか。それにLSI などハードウェアの技術もつながれば、応用としてニューロコンピュータの技術にもつながる。

21 世紀にこういう学問が確立するためには今から準備をしていかなければいけない。今われわれは 10 年かけて少しずつ体系化し、応用可能性を確かめていくための準備を 21 世紀に向けていまから開始すべき時

期にちょうどきたんじゃないだろうか。1960 年代に多くのエンジニア、科学者が決断をしてニューロの方では当面技術的可能性がないということで、逐次型の情報処理に全力をあげるべきだということになった。

いまもう一度振り向いて、並列の技術を取り入れる決断をすべき時期にある。

将来は全部がニューロ技術のほうに向かうかという、そんなばかなことはないので、人間の脳の中でも並列のダイナミカルな相互作用による情報処理と、直列のシンボルによる情報処理をやっている。

高次の知的な情報処理を実現するために、われわれはシンボルを発明し、シンボルを言語として定式化して逐次型の情報処理ができるようになったわけです。しかし人間はシンボル系列の逐次型の情報処理一本やりでやっているのかというと、そうではなくて、その下部構造に無意識下での並列のダイナミックスをもっていて、その並列のダイナミックスと意識下の逐次型の情報処理との両方を非常にうまい形で融合して使っているわけです。将来の技術体系はこういう方向へ向かっていく。この二つを融合するのに、現在われわれは並列型の情報処理の原理的可能性という意味では非常に弱いわけでありまして、いま、1990 年代からこの分野に力を入れて、逐次直列型と肩を並べて両方の技術を交流させ、融合させるようなものに、新しい学問体系を築いていく必要がある。

それが実は「ニューロコンピュータの展望」となるわけでありまして、いまブームであっても来年ニューロコンピュータができるとか、再来年できるとか、そんなことには絶対ならない。

印象的なのは、アメリカでは一方ではベンチャビジネスがニューロコンピュータと宣伝していますけれども、それはそれとして、MIT もそうだし、カルテックもそうですが、やっぱり 21 世紀に向けてそういう人材をいまから養成しなければいけないということで、大学院のカリキュラムのプログラムを全部そういう方向に変えてるわけです。そういう人たちが育って初めて学問体系が築けるので、ニューロコンピュータブームというのも結構だけれども、それに浮かれていないでしっかりした新しい学問体系と、それを担える人材をつくるようにいまから準備を固めておくべきではないか、こんなふうに思っているわけです。

どうもありがとうございました。(拍手)