

## AI の各種言語やツールと業務システム

日大生産工 ○豊谷 純 日大生産工 渡邊 昭廣  
日大生産工 若林 敬造

### 1. まえがき

本報告ではAIを実装するためのプログラミング言語やツールを解説して、企業内の基盤システムとの親和性を踏まえた全体構成を提案する。

研究背景として、現在は様々なシステムにAIが導入されて、業務の自動化が進められている。現在ではAIのライブラリも豊富であることから、Pythonが利用される事が多い。しかしながら電子決済やデータベースなど、いわゆる企業向けのシステムを構築する際には、JavaやF#に軍配が上がる。それらには開発の生産性や品質を高めるためのツールやライブラリが豊富に準備されており、何よりも開発技術者達がJavaやF#に慣れているためである。

コンピュータが誕生してから、その時代において必要とされる機能がミサイルの弾道計算から始まり、給与計算、Web販売システム、AIによる自動管理システムなど、多種多様に広がり、それに対応して作られて来た。

従って、本報告ではこのような状況下で、AIの変遷や種類、そしてプログラミング言語の種類について分類を行って全体を俯瞰する。そして実装例として、業務システムにAIを組み込む場合には、どのような構成でシステムを設計すれば良いかを考察する。

実装例として、ロボットによる自動倉庫の制御システムにAIを導入した場合に、どのような制御システムと、AIそれぞれに、どのようなプログラミング言語でシステム構成を行うのが良いのかを、システムの実装を踏まえて報告する。

### 2. 過去にAIが注目された経緯

AIが近年になって脚光を浴びているが、過去にも何度かのAIブームがあった。その当時はコンピュータの性能が実用に追いついておらず、限界が見えてブームが終焉したが、それらをまとめると次のようになる。

第一次ブームは1950,60年代で、主に「推論と探索」が行われ、Lispというプログラミング言語でパズルや迷路を解くことで成果を上げた。しかし、その頃のコンピュータの性能は低く、ルールとゴールが厳密に決まっている枠組のなかでしか動かせないため、現実世界では全く役に立たないことが分かり、その後は注目されなくなった。

第二次ブームは1980年代で、Prologというプログラミング言語で専門家の知識をコンピュータに教え込みことで、現実の複雑な問題を人工知能に解かせる「エキスパートシステム」に注力した。コンピュータは第1次ブームに比較すれば格段に高速になってはいたが、知識を教え込む作業が煩雑であり、業務では例外処理も多く、柔軟に対応させる事が難しくブームは去っていった。

第三次ブームは2000年代で、エキスパートシステムの問題を解決するために、「機械学習」や「ディープラーニング」の研究が行われた。2012年には画像認識技術で、「猫」を特定できるようになり、2017年にはスマートフォンで顔認証システムが実現された。過去のAIのブームではコンピュータの性能が追いついておらず、実用化に限界が見えてブームが去っていったが、この段階ではコンピュータが実用に耐え得る性能を持っているため、爆発的なブームとなっている。

### 3. AIとプログラミング言語の関係

通商産業省(現経済産業省)が1982年に立ち上げた国家プロジェクトの中核組織であるICOTの活動時期に対応して1990年代前半に掛けて人工知能ブームがあった。そしてLISPマシンによる医療情報エキスパートシステムで成果を上げた。その中で人工知能のエキスパートシステムを構築する際に、比較的容易に習得が出来るとして世界的にもPrologが注目された。

情報化の大きな波の中でデータが増大し、処理内容が具体的に細かく細分化が求められるようになると、Prologは処理が遅く注目されな

---

Programming language and AI tools and business systems

Jun TOYOTANI, Akihiro WATANABE and Keizo WAKABAYASHI

なくなった。そしてPrologがクラス概念を持たないために、マイクロソフト社による .NET アーキテクチャの共通言語基盤 (CLR) の対象言語から外された。そして近年、データサイエンスが脚光を浴びるようになると、元々は海外で普及して統計処理、人工知能関連等のライブラリが充実しているPythonが採用されるようになった。主なものを挙げると次のようになる。

#### Python

既に多数のデータサイエンスやAIのライブラリやパッケージがある。初心者にも使いやすく、多くのデータサイエンスのサポートツールがある。

#### Java

多くの企業で使われており、主にバックエンドのシステムやGUIやJava OpenGL (JOGL) 他によるデスクトップアプリに用いられている。

#### C++

他の言語よりも高速に動作するため、時間を要する処理の実行時間を短縮することが出来る。

#### JavaScript

サーバー側でデータを学習し、ConventJS(ディープラーニング)、Synaptic(ニューラルネットワーク)、Mind(訓練データを処理)などのライブラリを利用する。クライアントからはAjaxを利用して値を受け取って予測する。

#### Haskell

関数型プログラミング言語で、非正格な評価を特徴とする。

#### Julia

2012年2月にオープンソースとして公表されたプログラミング言語で、C言語などの静的型付け言語に迫る速い実行速度が特徴である。

#### Prolog

論理を使ったプログラミングを意味するフランス語を語源とし、第二次AIブームで利用された論理型言語である。

#### Lisp

AIを提唱したジョン・マッカーシーによって開発された言語で、第一次AIブームで利用された。

#### その他

開発言語では無いが、現在ではGUIによるAIツールが登場しており、プログラミング無しで、分析が出来るようになっている。

#### Neural Network Console

#### ソニーグループのGUIによるAI開発ツール MatrixFlow

#### MatrixFlowによるGUIによるAI開発ツール Watson Studio

#### IBMによるGUIによるAI開発ツール

### 4. プログラミング言語の利用形態による分類

コンピュータでの実行に際して各種言語を利用形態から分類すると、アセンブリ言語、コンパイラ型言語、インタープリタ型言語に分けることができる。ここでアセンブリ言語は、CPUのレベルに近い命令記述がとられることから機械語に近い言語で、低級言語とも呼ばれる。

コンパイラ型言語は、CPUは全く意識する必要が無く、人間が読みやすい形式で記述されており、変数名や繰り返し制御など、より抽象的な表現で記述されている。そして実行に際してはコンパイラを利用して機械語や中間言語などのより低レベルの命令に変換してから行う。インタープリタ型言語は、実行時に上から順番に、プログラムを読み込んで、逐一、命令の解釈と実行を行う。これはコンパイルの過程がないか、隠されている形態が取られている。

アセンブリ言語に対して、コンパイラ型言語とインタープリタ型言語は、人間に近い表現を用いるため高級言語と言われる。

### 5. プログラミング言語のパラダイムによる分類

どのようにプログラムを構成するかという観点(パラダイム)から分類すれば、手続き型言語、関数型言語、論理型言語、オブジェクト指向言語などが存在する。例えばオブジェクト指向ではオブジェクト(もの)の組み合わせで目的を達成させる。

手続き型言語は、古典的ではあるが行いたい処理を順番に記述するもので、Fortran, COBOL, PL/I, BASIC, C言語, Pascalなどがある。

関数型言語においては、関数の組み合わせによって記述をするもので、Lisp, Scheme, Haskellなどがある。

論理型言語では、記号論理学の体系を言語として実現したもので、Prologなどがある。

オブジェクト指向言語では、オブジェクト(もの)とオブジェクトの相互作用で目的を達成するように記述され、Simula, Smalltalk, C++, Objective-C, Java, C#, D言語などがある。

その他にインタープリタ型言語の実行形態のスクリプト言語があり、Perl, Python, JavaScriptなどがある。

## 6. 業務システムへの実装事例

### 6.1 エクセルによる管理業務にAIを導入

現在、AI化が進んでいるが、Pythonなどのライブラリを使ったものが大半である。しかしながら、会社の管理業務はエクセルで管理している場合が非常に多い。そこで、エクセルのVBAを利用して、著者がニューラルネットワークをVBAで作成した例を紹介する。

下記は過去のプロジェクトを分析して、機械学習をさせて、新規プロジェクトを実施した際の、収益の予測を行うAIプログラムである。

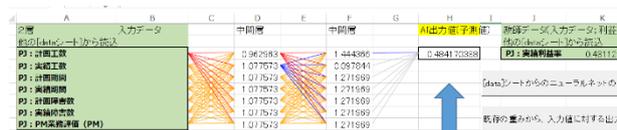


図1 エクセルでの実行画面(2層のみ)

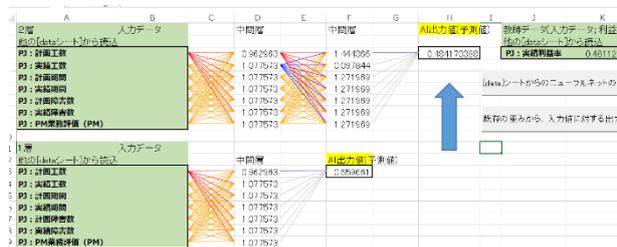


図2 エクセルでの実行画面(2層+1層)

表1 評価項目の例

PJ: 業種	PJ: PMコンピテンシー1
PJ: 業務	PJ: PMコンピテンシー2
PJ: 工程 (開始)	PJ: PMコンピテンシー3
PJ: 開発環境	PJ: PMコンピテンシー4
PJ: OS環境	PJ: PMコンピテンシー5
PJ: 基盤環境	PJ: PMコンピテンシー6
PJ: DB環境	PJ: PMコンピテンシー7
PJ: 計画工数	PJ: PMコンピテンシー8
PJ: 実績工数	PJ: PMコンピテンシー9
PJ: 計画期間	PJ: PMコンピテンシー10
PJ: 実績期間	PJ: PM業務評価 (PM)
PJ: 計画障害数	PJ: 計画利益率
PJ: 実績障害数	PJ: 実績利益率

### 6.2 営業所の最適配置をAIで計算

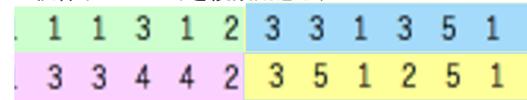
ここでは営業所の最適配置問題を例に挙げる。既に存在する営業所の緯度経度データを利用して、AIによってグループ分と物流センタ

一の最適立地を求める。AIの中でも具体的には遺伝的アルゴリズムを適用した。これは生物学における進化の過程で、遺伝と適者生存による自然淘汰の仕組みをソフトウェア的に模倣する事で複雑な問題に対する最適解を探索する手法である。

ここではトラックや荷扱い量の制約で、全体で5グループに分けた時の結果が図4、図5に示されている。

全店舗の緯度経度データを基に、店舗の密度関数を最小二乗法でモデル化する事により、扱う情報量を大幅に削減する事ができる。

- ある命題に対する解の候補を遺伝子 (gene) とその集合体である染色体 (chromosome) で表現した個体 (individual) を複数用意し、



- 適応 (fitness) の高い個体を優先して交叉 (crossover)、突然変異 (mutation) などの操作を繰り返しながら最適解の探索を行う。

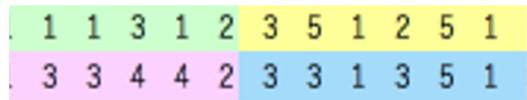


図3 遺伝子 t 交叉のモデル例

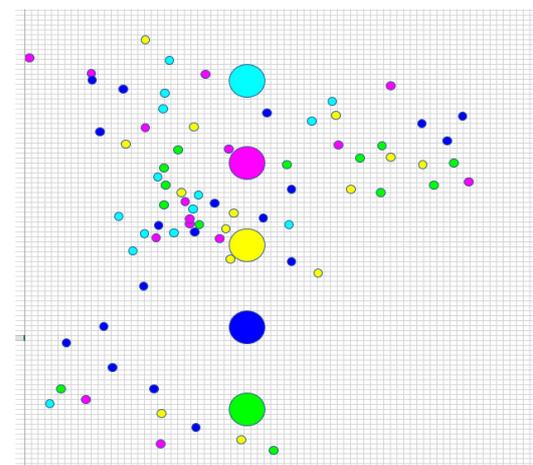


図4 初期のグループ割当

### 6.3 自動倉庫制御システムにAIを導入

近年の自動倉庫では、AGV(Automated guided vehicle)という無人搬送車が倉庫内を動き回って、荷物を移動するものがある。そのようなシステムの場合にも、荷物の配置場所や取り出す順番によって入出庫の作業効率は大

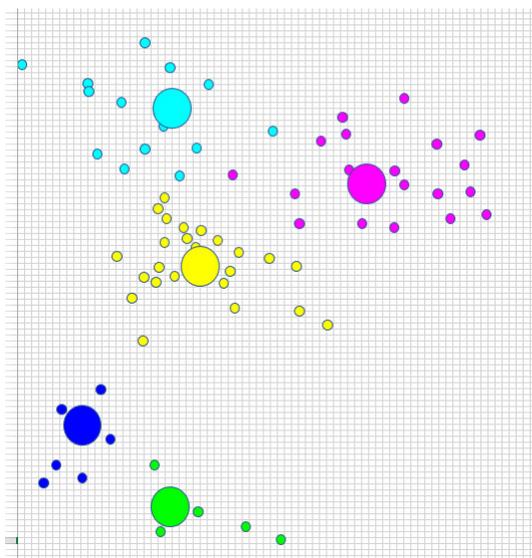


図5 最終的なグループ割当

ここでは、倉庫の荷物の配置場所を可視化する必要があったため、3Dグラフィック(OpenGL)とAIライブラリ、そしてデータベース(SQL)に対応した言語を検討した。そして特にそれぞれの技術資料がWebに豊富に存在することから、Javaを採用することにした。

JavaでAIを実装する際には、Deeplearning4jというDeep LearningのライブラリやJava-ML(Java Machine Learning Library)などの機械学習のライブラリがある。

図6は、ランダムに荷物を配置した結果で、図7は入出庫の過去の頻度を基に、AIによる最適配置を行って66.7%時間を削減した例である。

#### DirectX

マイクロソフト社のWindows環境のAPIで、ゲームなどの開発に利用されており、主なプログラミング言語はC++が多い。

#### OpenGL

多くの動作環境下でグラフィック全般を扱うためのAPIで、MACやUnix, iOS, Androidでも動作する。主なプログラミング言語はC/C++が多く、JavaにもJava OpenGL(JOGL)として対応している。

#### 7. おわりに

今回はAIを業務に利用する際のプログラミング言語について事例を挙げて紹介をした。Pythonでも作れたかもしれないが、プログラミング言語は仕様変更も多いため、利用する全

技術に対して十分な情報の整ったプログラミング言語の採択が求められる。

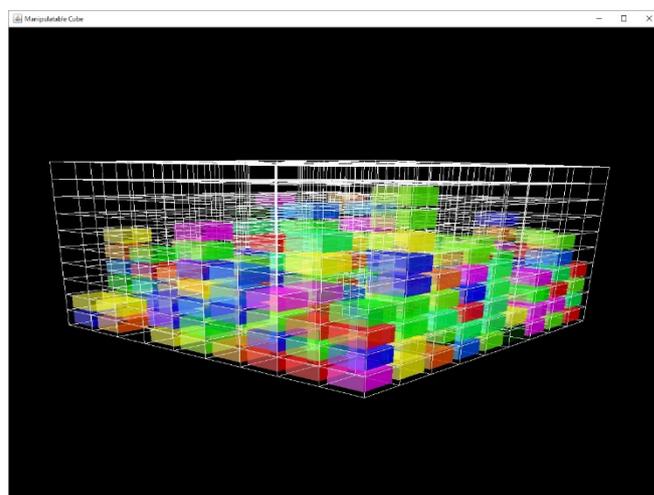


図6 ランダムな配置

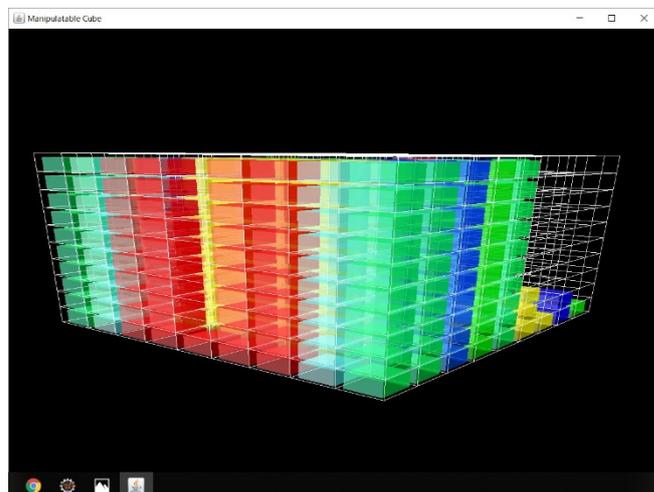


図7 入出庫の頻度を基に配置

#### 参考文献

- [1] 豊谷他, “経営のためのAIとプログラミング言語”, 日本情報ディレクトリ学会第23回全国大会研究報告予稿集, 2019年, P. 11-14.
- [2] 豊谷他, “AIによる店舗の自動グループ分け問題”, 日本情報ディレクトリ学会第22回全国大会研究報告予稿集, 2018年, P. 19-20.
- [3] 豊谷他, “AIを導入したソフトウェア開発の品質管理”, 情報処理学会, 第79回全国大会講演論文集, 6A-01, 2017年