

特集 「ゲーム産業における人工知能」

ビデオゲームと AI は相性が良いのか？

The Game and Artificial Intelligence Is the Good Compatibility?

森川 幸人
Yukihito Morikawa

株式会社ムームー
muumuu co.,ltd.
morikawa@muumuu.com, <http://www.muumuu.com/>

Keywords: artificial intelligence, game, etc.

1. はじめに

昨今の AI ブームで、ビデオゲームに AI を使いたいという話をよく聞きます。しかし、具体的にどう使えばよいか、どういう利点、欠点があるのか？ ビデオゲームという遊びと AI は、果たして相性が良いのか？ などの問題は実際に使った者でないとなかなか理解しにくいと思います。

過去に実際にビデオゲームに AI を使った経験を踏まえて、その点について考察したいと思います。

2. ビデオゲームで AI が活躍できる場

ビデオゲームで AI が活躍できる場は、大きく分けて三つの場が考えられます。

- 1) ユーザの動向判断や目標達成度の確認
- 2) パラメータなどのデータの設計
- 3) キャラクタの推論、行動



図1 AIが活躍できる三つの場

1) は、いわゆるデータマイニング技術で、今最も AI がうまく活用されている分野です。ビデオゲーム開発の現場で、ここに AI が積極的に利用されているという話は聞きませんが、ハードウェアの進化とともに、どんどんゲームの規模が大きくなり、複雑になり、開発コストがはね上がっている昨今、ユーザの動向解析や、いわゆる KPI（重要業績評価指標：Key Performance Indicators）の分野で、今後、AI が活躍するだろうことは容易に想像できます。

2) は、例えば、レースビデオゲームでのプレイヤーの車

以外の車のコース取りやシューティングゲームなどでキャラクターが進む経路を設計したり、RPG などでの武器やモンスターの強さや特性、アイテムの値段などのパラメータ値を設定することを意味します。こうした作業は従来、手作業で行われてきました。ビデオゲームでは、パラメータ類のバランスの善し悪しが、そのままビデオゲームの善し悪しにつながり、ソーシャルゲームでは、売上げにダイレクトに影響を与えるため、その設計には熟練したゲーム職人が必要でした。しかし、ビデオゲームが複雑化し、肥大化するに従って調整すべきパラメータが増えてきているため、優れたゲーム職人でも設計管理するのが難しくなっています。こうした作業の「コッ」を AI に学習させ、AI に設計させることは、ビデオゲーム開発のコストやリスクを大きく下げることが期待できます。



図2 キャラクタ、世界のパラメータ

ビデオゲームにおける AI の利用で、最も有望なのが、このビデオゲーム開発における AI の利用であると考えます。

3) は、最もビデオゲーム AI らしい使い方と考えられますが、後述するように、案外一番難易度が高いのではないかと思います。

2) のように、開発中に AI を利用する場合、AI がいかに大きなメモリや処理能力や学習時間を必要としても、開発コストがそれを許す限り問題ありません。また、

学習が収束しない場合も、再度学習させてよいので問題ありません。学習データや教師データや評価関数の精度も逐次修正して、精度を上げていくことができます。

これに対して、ビデオゲーム中に AI を学習させるには、次のような障害があります。

3. ビデオゲーム中に AI を使うときの困った問題

ビデオゲーム中に AI を使うとき、以下のような点が問題となります。

特に、b) ～ e) は、ビデオゲームという遊びの本質からの要請であり、デバイスの性能が上がっても解決しません。

また、不幸にも、AI にとってはどれも都合の悪い要請であるといえます。

- a) デバイスのスペックが低い
- b) 必ず、「正解」に達しなくてはならない
- c) 短い時間で学習を終了しないといけない
- d) 学習速度が良いテンポでないといけない
- e) 追加修正学習をしないといけない

3.1 デバイスのスペックが低い

高性能の PC 用に提供されるゲームでは問題ないかもしれませんが、提供されるデバイスが、ゲーム専用機やスマートフォンの場合、デバイスのメモリ量や CPU のパワーが問題になります。

特に、計算負荷が高く大きなメモリを必要とするニューラルネットワークを使う場合には、パワー不足が大きな障害となります。

3.2 必ず、「正解」に達しなくてはならない

ほとんどのビデオゲームの場合、ゲームクリアのための「正解」があります。

例えば、「XXX を見つけて倒す」という目的があったとします。そのためには、必要なスキルやアイテムを身につけ、目的地や攻撃法などの情報を正しい手順と正し



図3 どのイベントも順番どおりに行わないといけない

い手続きで手に入れ（それによって、XXX に会えるフラグが立つ）、万全の準備のもと、XXX に立ち向かうことになります。

この一連のイベントはどれも省くことができず、取得する順番も正しくなくてはなりません。AI は内部に乱数を利用することが多いなどの理由のため、必ずしも「正解」に達することが保証されません。このため、もし、このような行動判断を完全に AI に任せるとすると、誤ったルートを進んだり、間違った攻撃をしたり、必要な情報を手に入れられなかったりして、イベントをクリアできない可能性が出てきてしまいます。

3.3 短い時間で学習を終了しないといけない

AI、特にニューラルネットワークモデルを利用する場合は、その学習には多くの時間が必要となります。2) のように開発中であれば問題ありませんが、ビデオゲーム中に学習させるとなると問題になります。ビデオゲームは静止を嫌います。バグに見えたり、世界観を壊したりするだけでなく、プレーヤが自分のタイミングでゲームが進まないことを快く感じないためです。AI が学習中であろうと、プレーヤのタイミングで「進む」必要があるので、AI の学習も瞬時に完了する必要があります。そのため、AI を学習させるタイミング、AI モデルの規模、ニューラルネットワークモデルでいえば、ノードやレイヤの数、教師信号の数、求める学習精度などについて、細かなチューニングが必要となります。また、時として解決不能な問題となります。

3.4 学習曲線はゲーム的カーブを描かないといけない

一般に、AI の学習の収束は、早ければ早いほど良いものですが、ビデオゲームの場合は、そうとも限りません。AI の学習曲線は一般的に、序盤に一気に学習が進み、やがて踊り場に達して、その後、緩やかに学習が進んでいきます。これをビデオゲーム的に訳すと、序盤で一気に賢くなった後は、ゆっくりまったり賢くなっていくということになります。これでは、ビデオゲーム的には具合が良くありません。序盤も中盤も終盤も同じように「成長」していく、あるいは、後半に向かってどんどん加速的に成長していくようでないビデオゲーム的につまら

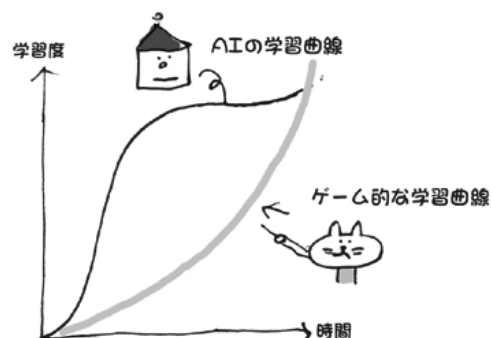


図4 ゲーム的に理想の曲線

ないものになってしまいます。このため、AIの「自然な」学習の収束タイミングでは具合が悪く、AIの成長速度をチューニングしていく必要があります。

3.5 追加修正学習をしないとイケない

プレイヤーの判断や方針をAIの教師信号として利用する場合は、教師信号の追加という問題が発生します。あらかじめ、すべてのイベントについて学習させておくことはできませんから、プレイヤーの判断や方針を反映させようとすると、どうしても追加学習が必要になります。それも度重なる追加学習が必要となります。ニューラルネットワークモデルでは、そもそも、追加学習させること自体があまり具合の良いものではなく、往々にしてそれまでの学習を破壊してしまいます。

また、プレイヤーがビデオゲームを進めるうちに判断や方針がぶれることがあります。このため、矛盾した教師信号が追加され、教師信号の精度が下がっていきます。追加学習と教師信号の精度の低下という二重の要因により、学習がうまく収束しない可能性が高くなります。

以上のような理由から、ビデオゲーム中にAIを利用する、特に学習させる場合は、多くの問題が発生します。これらの問題が発生する頻度やリスク度は、ゲームごとに違うため一般的な対処法を確立しづらいのも問題です。

20年前からビデオゲームにAIを利用してきた著者もこの問題に直面しました。そして、試行錯誤の結果、以下のような対処を行いました。

以下では実際にAIを利用したゲームにおける具体的なAIの利用方法、問題に対する対処法を記します。

4. 「がんばれ森川君2号」^[プレステ97]

プレイステーションは、2メガビット！のメモリしか乗っておらず、CPUのパワーも貧弱であったため、ニューラルネットワークモデルを使うには全く不都合なマシンでした。しかし無知のなせる技、無謀に



図5 「がんばれ森川君2号」
© 1997 Sony Computer Entertainment Inc.

も、当時ニューラルネットワークモデルの最先端であった「バックプロパゲーション」という学習アルゴリズム(Backpropagation, 以後BP)を使うことにしました。このゲームは、世界にあるいろいろなアイテムに接触しながら、それが食べ物か、危険なものか、楽しいものかなどを学習していくというゲームで、プレイヤーは、キャラクタにそれを教えるという飼育主的立場となります。

今でいえば「育成+放置ビデオゲーム」に該当するかと思います。この学習にBPを利用しました。入力は、アイテムの形状や動き、におい、音などの視覚、聴覚、嗅覚に関わるデータ、出力はアイテムのカテゴリとなります。

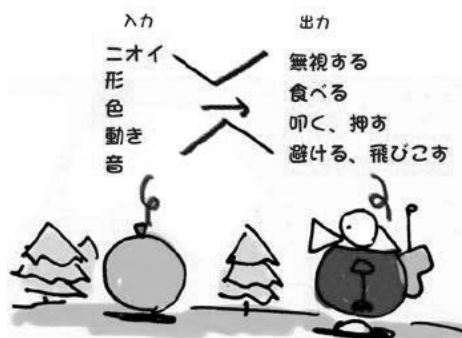


図6 ニューラルネットワークの入出力要素

上記のようなマシンスペックであり、3章のb)～e)のビデオゲーム的な要請を満たすため、学習に使う時間を1秒以内に納めることを目標にノード数を調整しました。学習がいつも同じ割合で進んで見えるように、学習のエラー値を監視し、エラー値の減少が一定になるように学習回数を逐次調整しました。それでも学習が進まない場合は、キャラクタが「寝ている」間に学習を進ませました。また、万が一、学習が収束しなかったとき(プレイヤーの教えをちゃんと覚えられない場合)のため、バックグラウンドで、バックアップ用のBPを学習させ、いよいよ学習が収束しないとなると、バックアップの中でうまく学習しているものとこっそり入れ替えました。このバックアップのAIの学習も、キャラクタが「寝ている」間に行われました。つまりキャラクタの睡眠は、単に生き物らしい演出ではなく、人間の睡眠と同様に「脳の情報整理」のための必然的な生理機能なのでした。

このゲームでは、どんどん新しいアイテムに出会うため、追加学習が必須となりました。この追加学習による学習精度の低下を避けるため、それまでに教えたことと新たに追加されたことの学習回数に重みを付けた教師信号を生成しました。例えば、それまでに教えたことを1回繰り返すのに対して、新たに教えたことは5回繰り返して学習させる、つまり、それまでの教師信号1セット+新しい教師信号5セットで新たな教師信号をつくるというイメージです。こうした対処で過学習を回避し、古い学習と新しい学習のバランスを取りました。

5. 「アストロノーカ」^[プレステ 93]

「アストロノーカ」は、宇宙一の作物を育てることを目標とする宇宙農家になるというビデオゲームで、作物を育てている間に、それを狙う害獣「バブー」がやって来るため、プレーヤは、トラップを仕掛けてバブーを撃退します。



図 7 「アストロノーカ」

© 1998 MuuMuu Co., Ltd. / SYSTEM SACOM Corp. / SQUARE ENIX CO., LTD.

このバブーのトラップ突破の行動判断や身体パラメータを遺伝的アルゴリズム (Genetic Algorithm, 以後 GA) によって進化 (学習) させました。

バブーは、トラップごとに、避けて迂回すべきか、飛び越すべきか、壊すべきかなどの対処法を学習します。同時に、ジャンプ力を高めるために脚力を上げたり、破壊力を増すために体重を増やしたり、ダメージに対する耐性を高めたりするなど、身体パラメータを最適化していきます。例えば、 $\text{ジャンプ力} = \text{脚力} / \text{体重}$ 、 $\text{破壊力} = \text{腕力} \times \text{体重}$ 、といったように、各パラメータがトレードオフになるように設計しました。より高くジャンプするには、脚力を上げるとともに体重を下げる必要があるが、それでは破壊力が落ちてしまうことになります。必ずしもすべてのパラメータ値を上げるだけではうまく適応できません。バブーは、プレーヤの配置するトラップ種に合わせて、各パラメータをうまく調整するように進化します。

プレーヤは任意のトラップを任意の個数配置することができましたので、従来のビデオゲームのように、事前にプレーヤの「強さ」に応じてキャラクタのパラメータを設定するということできません。GA によって自律的に最適なパラメータを獲得していく仕組みは、大変具合の良い手法でした。

しかし、GA もまた BP 同様、適応が初期に極端に進んだ後、踊り場 (まったりとした進化) に至り、ゲーム的に気持ち良い進化スピードになりません。また、GA の場合、かなりの確率で、局所解に陥ってしまっ



図 8 バブーの回避行動の進化

適なパラメータの組合せにならなかったり、うまい回避方法を取得できないまま、進化の袋小路に入ってしまう。

それに対して以下のような対処法をとりました。ゲーム的には、毎日、バブーがやってくるという演出になっていましたが、内部では、平均 20 世代交代させていたのですが、内部では、平均 20 世代交代させた後、最も成績の良い個体を選ばれ出撃していく仕組みになっていました。この際、各世代の適応度を常に監視して、世代数は適宜に変更しました。学習が進みすぎる場合は、世代交代を減らして出陣させ、学習が進まない場合は、20 世代以上の世代交代をさせた後出陣させました。また、通常はランキング方式で親を選び、進化が滞ってくると、ルーレット選択に変更したり、突然変異率も適宜に変化させるなど、動的に GA の仕様を変化させました。また、プレーヤのトラップでは太刀打ちできないくらい進化してしまう (ハマリ状態になる) ことを想定して、GA の評価関数に「同じ成績なら、パラメータ値が低いほど有利」という要素を加え、例えば、同じようにトラップを突破した場合は、パラメータの低い個体のほうが選ばれやすくなりました。結果、バブーが手に負えなくなったらトラップを置かないで素通りさせてしまうことで、集団が弱体化 (退化) し、「ハマリ」を解消させました。

こうした対処により、GA によってキャラクタをゲーム的に気持ちが良いテンポで、うまく進化させることができました。

バブーのパラメータは、初期集団の 20 個体分を設計したのみで、無限に近いほかの個体のパラメータは、GA が生成しています。ゲーム製作の段階においても、

GAは大変有効でした。

通常のゲームのつくり方では、キャラクターのパラメータはすべて個別につくります。100体のキャラクターがいれば、100セットのパラメータをつくります。パラメータは、そのキャラクターが登場する状況やタイミング、つまり、プレーヤのレベル、装備、所持金、ショップでの売り物、そのキャラクターが生息するエリアなどに対応させて設計する必要があります。

この状況とのバランス調整は大変難しい作業です。作業量が膨大であり、制作中に状況が変更になれば一からつくり直さなければなりません。そのうえ、このバランス調整がゲームの善し悪しを決めてしまうため、センスが問われます。それでも、あらかじめすべての状況がわかっている場合は、力業で設計することは可能ですが、ゲームデザインとして、プレーヤの選択に高い自由度があると、そのキャラクターに遭遇する状況が想定しきれなくなり、パラメータ設計は極めて難しい作業となります。

「アストロノカ」においても、プレーヤが配置するトラップ種と個数、配置場所などは想定できません。しかし、バブーの進化をGAで行ったため、初期集団の20体のパラメータのみを設計しただけで済みました。ゲームクリアまでにどれだけのバブーが登場するかは、ユーザによって異なるため定かではありませんが、仮に100体であったとしても、パラメータ設計の作業は1/5になったわけです。

このように、AIはゲーム内のキャラクターに利用するだけでなく、ゲーム開発の段階においても、大変有効なツールとなります。

6. 「くまうた」^[プレステ 03]

「くまうた」^[プレステ 03]は、宇宙一の演歌歌手を目指す(宇宙)クマが、師匠(プレーヤ)のところで演歌の作詞作曲、歌唱を学ぶというゲームです。

歌唱は当時としては珍しかった音声合成技術を利用しています。「がんばれ森川君2号」^[アストロノカ]に



図9 「くまうた」
© 2003 Sony Computer Entertainment Inc.

続き、このゲームでも、ゲーム内キャラクターの「リアリティー」を上げるためにAI的な仕組みを利用しました。

キャラクターの「リアリティー」は、モデリングやテクスチャ、モーションといった見た目のリアルさを追求していても到達できないだろうと考えていました。キャラクターが自ら考え、自らメッセージを発信するといった「知能」がないとダメであり、逆に、それがあつたら、ただのキューブでさえ、生き物感を感じてくれるはずだと考えました。

自らメッセージを生成するという事は、事前に音声を取録できない、つまり、ゲーム中に音声を合成する必要があります。しかし、当時の音声合成技術は、抑揚のない機械的な発声しかできなかったため、これではかえってキャラクターのリアルさを失ってしまう危険がありました。あるときラップ音楽を聴いていたとき、ふと、音楽に乗せれば、イントネーションが間違っている、抑揚がなくても、それらしく聞こえるのではないかと気が付き、キャラクターのメッセージを歌に乗せることを思い付きました。こうして、歌に乗せて、キャラクターに自然にしゃべらせたいというアイデアから逆算してつくったのがこのゲームです。

音声合成で歌を歌うことではボーカロイド「初音ミク」が有名ですが、初音ミクのリリースは2007年、「くまうた」は2003年ですから、少し先輩にあたります。

そして、どうせならと、曲も歌詞も自動生成することにしました。曲の自動生成は、一般に無個性な曲になりがちのため、特徴が明確に出るジャンルを選ぶ必要がありました。具体的には、ラップ、ブルース、'70フォーク、演歌などがそれに当たり、結果、演歌を選びました。

かわいい女の子×テクノポップの組合せに対して、クマ×演歌という世界観を選んだところに、著者の商才のなさがうかがえます。

曲や歌詞の自動生成は、当初、GAを使う予定でいましたが、歌詞全体を通して意味が通じるように学習させるのが困難だったため、穴あきのテンプレートにランダムに言葉を代入し、プレーヤがその選択を評価する、または、直接言葉を与える仕組みにし、クマはプレーヤの教えや評価をもとに言葉選びを学習していきます。

メロディの生成も同じような仕組みで自動生成しました。自動生成された演歌は、なかなかもつとらしいものになり、バリエーションも豊富になったのですが、そもそも、演歌は何を聴いても同じに聞こえるというのは盲点でした。

7. 「てきとうパパ」^[てきとう 13]

「てきとうパパ」^[てきとう 13]は、子供の質問(=プレーヤの選択)に対して、お父さんがそれを説明する。最初は正しいことを言っているのがだんだんと脱線してヘンな解説になっていくのを楽しむというスマートフォ



図 10 「てきとうパパ」
© 2013 MuuMuu Co., Ltd.

ンゲームです。子供の質問として、プレーヤの自由なテキスト入力にも対応しています。

ウィキペディア (Wikipedia) をデータベースとして利用して、解説文のひな型に適当に選ばれた単語を組み込む形で、お父さんの解説文を自動生成していきます。

ビデオゲームに AI を使うとき、学習済みの AI を使う場合と学習過程の AI を使う場合が考えられます。キャラクターを育てる遊びの場合は、後者を利用します。学習過程をキャラクターの成長過程として楽しもう、間違いもまた可愛さのうち、という考えです。「てきとうパパ」では、この流れを逆にして、学習完了 (= 正解) から学習過程に戻る流れで、どんどんお父さんの話が「脱線」していくように演出しました。

このゲームではニューラルネットワークは使いませんが、ニューラルネットワークの学習過程での間違いは、人間の間違い方と似ている点があり、間違いに同感したり同情したりできるところがあります。正解にたどり着く、正解を出すのが、生き物らしいわけではなく、そこに至るまでが生き物らしいわけですから、学習過程をキャラクターの「味」に利用しない手はありません。

ただし、先の理由から、必ずしも正解にたどり着けないことがある点は要注意で、そうした場合の対処を考えておく必要があります。

8. ま と め

ビデオゲーム中で AI を使う場合は、どのジャンルのゲームであるにしろ、以上のような利点欠点があります。

「リアル」を求められるビデオゲームは、今までは、モデルやテクスチャ、アニメーションのリアルさでそれに応じてきていますが、本当の「リアル」を表現するためには、キャラクターの行動判断も「リアル」である必要があります。その「リアル」を表現するためには、AI は最適な技術と考えます。

また、ゲーム制作の場においても AI を使うことが有効です。

しかし、従来の AI モデルがビデオゲームに適しているかという点とそうでない気もします。小さいメモリやパワーの低い CPU でも動き、学習が素早くかつビデオゲーム的に良いテンポで進む、追加学習にも強い AI は、既存のモデルの中には見当たりません。ビデオゲームに本格的に AI を使うには、やはり、ビデオゲーム専用のチューニングされた独自の AI の開発が必要と考えます。

◇ 参 考 文 献 ◇

- [プレステ 97] がんばれ森川君 2 号：プレイステーション用ビデオゲーム, ©1997 Sony Computer Entertainment Inc. (1997)
- [プレステ 98] アストロノカ：プレイステーション用ビデオゲーム, ©1998 muumuu co., ltd., SYSTEM SACOM Corp., SQUARE ENIX Co., Ltd. (1998)
- [プレステ 03] くまうた：プレイステーション 2 用ビデオゲーム, ©2003 Sony Computer Entertainment Inc. (2003)
- [てきとう 13] てきとうパパ：スマートフォン用ビデオゲーム, ©2013 muumuu co., ltd. (2013)

2017 年 1 月 11 日 受理

—— 著 者 紹 介 ——



森川 幸人

グラフィック・クリエイター。1959 年岐阜県生まれ。1983 年筑波大学芸術専門学群卒業。株式会社ムームー代表取締役。主な仕事は、CG 制作、ゲームソフト、スマホアプリ開発。2004 年「くまうた」で文化庁メディア芸術祭審査員推薦賞、2011 年「スカカの結婚」で第 1 回タ・ヴィンチ電子書籍大賞大賞受賞。代表作は、「アインシュタイン」、「ウゴウゴルーガ」(テレビ番組 CG)、「ジャンピング・フラッシュ」、「アストロノカ」、「くまうた」(ゲームソフト)、「マッチ箱の脳」、「テロメアの帽子」、「スカカの結婚」、「絵でわかる人工知能」(書籍)、「スカカの結婚」、「アニマルレスキュー」、「ねこがきた」など (iPhone, android アプリ)。