

流行りの(?)ニューラルネットワークを使って各役職の位置を予測します。

発言や投票、襲撃といった行動については、自らの経験から定石と思われるものを実装しました。

定石で確定できない投票先や襲撃先が生じた時に、ニューラルネットワーク予測機とその他いくつかの予測機でそれぞれ役職を推定し、その合計で誰を指定するべきか判断します。

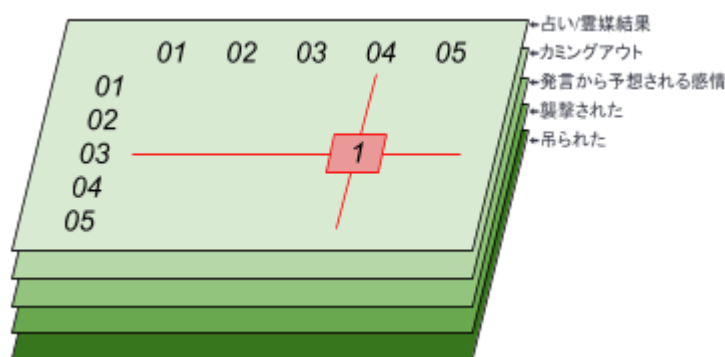
このレポートでは、ニューラルネットワークを使った予測機に焦点を絞って説明します。

## 実際にやったこと

### 村の状況を配列で表現

[村の人数] $\times$ [村の人数] $\times$ [使う特徴の数]  
の3次元配列を用意します。特徴は

- 占い/霊媒結果
- カミングアウト
- 発言から予想される感情
- 襲撃された
- 吊られた



の5つを考慮しました。

1ターン毎に、発言や行動に応じてこの配列を更新していきます。例えば「Agent[03]がAgent[04]を占ってHUMANと判定した」場合には、この配列の3行4列1層目に1と記録します。

### 配列から学習

この配列を入力、誰が本当の狼かを出力とするニューラルネットワークをKerasを用いて構築します。学習データには公開されている過去の決勝と今回の予備予選終盤のログを使用しました。

約4時間の学習の後、最終的な正答率は65%前後になりました。同様にして、狼だけでなく占い師、村人、狂人、(これに加えて15人村では狩人、霊媒師)と各役職毎に予測機を構築します。

### ゲーム内での運用

以上で作成した各役職ごとの予測機をエージェントに組み込みます。ゲーム中も同じように配列を更新していき、投票先や襲撃先を選択する場面で配列を入力し、「最も狼らしい人物」、「最も占い師らしい人物」などを受け取ります。

この時単にその役職の予測機だけを使うのではなく、全ての役職予測機に配列を入力し、例えば狼を予測する場合、狼である確率からその他の役職である確率を引いたものを使うことで、僅かに的中率が改善しました。

## 本来やりたかったこと

### 村の状況とグラフ構造

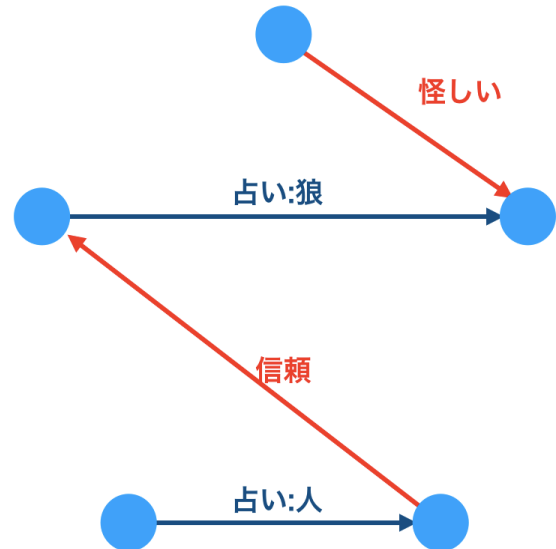
人狼における村の状況は、右図で示すように、各エージェントをノード、行った行動をエッジとする有向グラフで表現できそうです。

エッジにはいくつかの種類が想定されます。

- 占い/霊媒結果
- カミングアウト(常に自己ループ)
- 投票
- 発言から予想される感情

などなど...

「実際にやったこと」で説明した配列は、このグラフを種類ごとに隣接行列的な形で表現したものになります。



### グラフの分析

このようにして作成したグラフには、様々な使い道が考えられます。

例えば、良い印象(占った結果白だった、信頼している、一度も投票していない)のエッジだけを取り出したものは、仲の良いエージェント同士が繋がったソーシャルグラフとして見ることができます。これにクラスタリングアルゴリズムを適用することで、狼陣営らしいクラスタを炙り出せるかもしれません。

### 問題

この方法では、エッジに表現された行動がいつ行われたものなのかという情報が完全に欠落してしまいます。特に占い結果などはどちらが先に占ったのかという情報が鍵を握る場合も多いため、こういったタイミングについての情報を保持するにはどうすれば良いかが課題です。

また、そもそもこの項で述べたことが本当にできるのか、効果があるのか全く不明です。

本来はこういったことをしたかったのですが、時間と勉強不足により、上述した妥協バージョンで実装しました。次回参加時に改めて挑戦したいです。

### 最後に

TRKOkamiチームからはいくつかの便利な予測モジュールを提供していただきました。このレポートに記載した内容以外に使用したモジュールの詳細については、TRKOkamiチームのレポートをご参照ください。