OSTBAYERISCHE
TECHNISCHE HOCHSCHULE
REGENSBURG

INFORMATIK UND
MATHEMATIK

Comparative Analysis of Predictive Performance: Neural Networks vs. GRU in League of Legends Match Outcome Prediction

Bachelorarbeit
von

Moritz Palm
Matrikelnummer: 1234567

Fakultät Informatik und Mathematik
Ostbayerische Technische Hochschule Regensburg
(OTH Regensburg)

Gutachter: Prof. Dr. Brijnesh Jain
Zweitgutachter: Prof. Dr. Timo Baumann

Abgabedatum: November 3, 2023

Herr
Moritz Palm
Konrad-Adenauer-Allee 55
93051 Regensburg

Studiengang: Künstliche Intelligenz & Data Science

1. Mir ist bekannt, dass dieses Exemplar der Bachelorarbeit als Prüfungsleistung in das Eigentum des Freistaates Bayern übergeht.

2. Ich erkläre hiermit, dass ich diese Bachelorarbeit selbstständig verfasst, noch nicht anderweitig für Prüfungszwecke vorgelegt, keine anderen als die angegebenen Quellen und Hilfsmittel benutzt sowie wörtlich und sinngemäße Zitate als solche gekennzeichnet habe.

Regensburg, den November 3, 2023

Moritz Palm

Contents

1 Introduction

esports is highly relevant due to it being a huge and strongly growing market. many games are hard to understand, due to lots of information being displayed with very little explanation a win predicition graph can help viewers understand the action and the significance of certain plays better, thus increasing engagement and enjoyment. riot games has already implemented their own proprietary win prediction a win prediction model can also help players make more informed decisions about what the optimal path of actions is

2 Background

2.1 League of Legends

League of Legends is a Multiplayer Online Battle Arena (MOBA) game developed by Riot Games. MOBA games are a subgenre of real-time strategy games in which two teams, typically consisting of five players ('summoners') each, compete against each other with each player controlling a single character [1], called 'champion'. It is one of the most played video game genres CITATION NEEDED and attracts millions of players and fans watching the professional scene. Most MOBAs differ only slightly in terms of basic gameplay or map layout, but vary in details such as champions, abilities, graphics etc. As League of Legends is the most played game in the MOBA genre, we will focus on it. At the start of the game, each player picks a champion from a pool of currently 165 champions, each with distinct abilities and characteristics. The map consists of two bases which are connected by three lanes. Each base contains a large structure, the so called 'nexus', which is protected by two turrets. The goal of the game is to destroy the enemy nexus. The bases are connected by three lanes, separated by a jungle. Non-player characters (NPCs), so called 'minions', spawn in regular intervals and advance down the lanes. Killing minions grants gold and experience points (xp), which are used to improve different attributes by buying items or increasing ones level. It is conventional that the players split up at the start of the game, with one player going to the top lane, one to the mid lane and two players to the bottom lane. The last player gets his gold and xp from killing neutral monsters in the area between each lane, commonly referred to as 'the jungle'. It also contains two large neutral monsters, Baron Nashor and a dragon, which require multiple team member to be killed and grant improvements to the whole team. Larger fights are usually centered around either destroying a turret or killing a large neutral monster. Every year, the game enters a new 'season', where it undergoes major challenges. In between seasons, the developers release smaller patches every two weeks, which are usually aimed at changing the strength of different champions.

The players need to pick a champion that fits best into the team, taking into account the team strategy, the damage composition, the role the player has been assigned etc., while also selecting a champion the player has played before and can play well.

league of legends is a zero sum game

Esports and mobas in particular are hard to understand and follow. A live game prediction view can help fans understand the action and decisions made better and help immerse the audience by detecting upsets and swings in win probability.

patches can alter the game significantly, so it is important to train only(or mostly) on recent data

the model should be able to answer the question, if team a is far enough ahead to win or if team b with their hyperscaling heroes can come back and win

2.2 Neural Networks

Artificial Neural Networks (ANNs) have been originally designed to simulate the way the human brain processes information. The fundamental unit in an ANN is called a neuron, first proposed by McCulloch and Pitts [2]. A neuron is a function which takes a number of values $\mathbf{x} = (x_1, x_2, ..., x_m), m \in \mathbb{N}$ and outputs a single value $\hat{y}$. Each input x_i is individually weighted by the learnable weights $\mathbf{w} = (w_1, w_2, ..., w_m)$ and added to a bias term w_0: $z = w_0 + w_1x_1 + ... + w_mx_m = \mathbf{w}^T\mathbf{x}$. z is then passed through an activation function ϕ, which provides non-linearity: $\hat{y} = \phi(z)$. Common activation functions include sigmoid functions, Rectified Linear Unit (ReLU) and more. The Multi-Layer Perceptron (MLP), first introduced by Rosenblatt [3] organizes neurons in layers. Input data is fed into the input layer, and computations propagate forward through the hidden layers to produce an output. In the simplest form of an ANN, every neuron in a layer is connected to every neuron in the subsequent layer, forming a fully connected architecture. There are no other connections, making it a feed-forward network, and it constitutes a directed acyclic graph. To allow the network to approximate any measurable function, at least one hidden layer is required [4]. During the training process, the weights and biases are iteratively adjusted using the backpropagation algorithm to minimize the loss function E. For regression tasks, a commonly used loss function is the Mean Squared Error (MSE) Loss

$$E_N = \frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2 \tag{1}$$

where N is the number of samples [5]. For binary classification, the Cross-Entropy Loss (CEL)

$$E_N = \frac{1}{N}\sum_{k=1}^{N} y_k \ln \hat{y}_k + (1 - y_k) \ln (1 - \hat{y}_k) \tag{2}$$

is widely used [6].

2.3 Recurrent Neural Networks

Recurrent Neural Networks (RNNs) build upon the foundation of feed-forward neural networks. They are designed to handle sequential data by introducing the concept of recurrence. In a RNN, the output at each time step depends not only on the current input but also on the network's previous internal state or hidden state. This allows RNNs to capture dependencies over time, making them suitable for tasks involving sequential input such as speech and language [7]. The first fully connected RNN was proposed by Elman [8]. The sructure of the Elman Network Cell can be seen in REFERENCE MISSING. RNNs, however, have challenges with vanishing gradients, especially in long sequences, which can affect their ability to capture long-range dependencies [9]. To

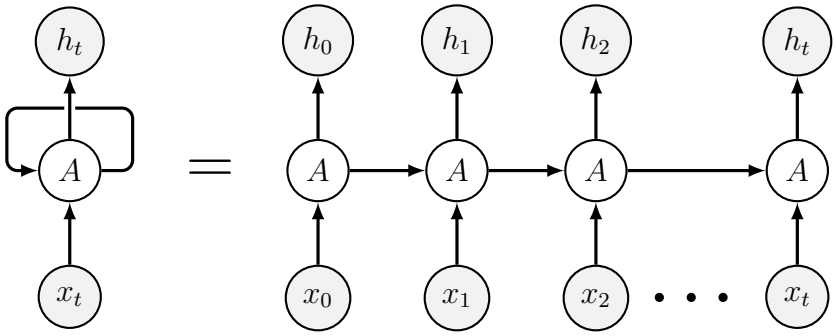

Figure 1: Unrolling of a RNN over time

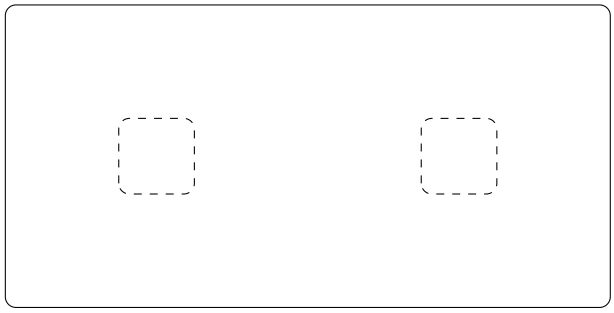

Figure 2: Elman Recurrent Unit

address this, more advanced RNN architectures, such as Long Short-Term Memory (LSTM) and Gated Recurrent Unit (GRU) networks have been developed. These architectures introduce gating mechanisms to mitigate the vanishing gradient problem and enhance the modeling of sequences.

2.4 GRU

Explain and define GRU, especially in comparison to LSTMs

2.5 Definitions and notations

2.6 Related work

Utilizing machine learning methods to extract information from data generated by e-sport games is an area of ongoing research. Most scientific research focuses on the similar MOBA DotA 2, which has easier and more fine-grained data collection methods (see section 3.1). Due to the high similarity between these two games, it is to be expected that any findings for one game can be replicated and used for the other game with minimal adaptations.

Silva et al. have used RNNs to predicting the winner using data of different time intervals. They achieved an accuracy of 75% when using data from between the 10 and 15 minute mark. An evaluation of LSTM resulted in lower accuracy, most likely due to

the large amount of data required [10].

3 Experiments

3.1 Data Collection

Most of the data is available from Riot Games API. For data not available directly from Riot Games, a web-scraping approach has been chosen. All of the data is saved first to a PostgreSQL-Database, from which the different datasets are constructed.

3.2 Combination of static and temporal data

4 Results

5 Discussion

6 Conclusion

Bibliography

[1] M. Mora-Cantallops and M.-Á. Sicilia. MOBA Games: A Literature Review. *Entertainment Computing* 26, 128–138, May 2018.

[2] W. S. McCulloch and W. Pitts. A Logical Calculus of the Ideas Immanent in Nervous Activity. *The bulletin of mathematical biophysics* 5(4), 115–133, Dec. 1, 1943.

[3] F. Rosenblatt. The Perceptron: A Probabilistic Model for Information Storage and Organization in the Brain. *Psychological Review* 65(6), 386–408, 1958.

[4] K. Hornik, M. Stinchcombe, and H. White. Multilayer Feedforward Networks Are Universal Approximators. *Neural Networks* 2(5), 359–366, Jan. 1989.

[5] K. Liano. Robust Error Measure for Supervised Neural Network Learning with Outliers. *IEEE Transactions on Neural Networks* 7(1), 246–250, Jan. 1996.

[6] Z. Zhang and M. Sabuncu. Generalized Cross Entropy Loss for Training Deep Neural Networks with Noisy Labels. In: *Advances in Neural Information Processing Systems*, 2018.

[7] Y. LeCun, Y. Bengio, and G. Hinton. Deep Learning. *Nature* 521(7553), 436–444, May 28, 2015.

[8] J. L. Elman. Finding Structure in Time. *Cognitive Science* 14(2), 179–211, 1990.

[9] I. Sutskever. "Training Recurrent Neural Networks". PhD thesis. Toronto, ON, Canada: University of Toronto, 2013. URL: `https://www.cs.utoronto.ca/~ilya/pubs/ilya_sutskever_phd_thesis.pdf` (visited on 11/01/2023).

[10] A. L. C. Silva, G. L. Pappa, and L. Chaimowicz. Continuous Outcome Prediction of League of Legends Competitive Matches Using Recurrent Neural Networks. *SBC-proceedings of SBCGames*, 2179–2259, 2018.

List of Figures

List of Tables

List of Abbreviations

ANN Artificial Neural Network

CEL Cross-Entropy Loss

FNN Feedforward Neural Network

GRU Gated Recurrent Unit

MLP Multi-Layer Perceptron

MOBA Multiplayer Online Battle Arena

MSE Mean Squared Error

NN Neural Network

NPC non-player character

ReLU Rectified Linear Unit

RNN Recurrent Neural Network

xp experience points