

BRENDON HENRIQUE DE PAULA DA SILVA

PREDIÇÃO DE VENCEDORES EM PARTIDAS PROFISSIONAIS DO JOGO *LEAGUE OF LEGENDS (LOL)*

(versão pré-defesa, compilada em 13 de dezembro de 2021)

Trabalho apresentado como requisito parcial à conclusão do Curso de Bacharelado em Ciência da Computação, Setor de Ciências Exatas, da Universidade Federal do Paraná.

Área de concentração: *Ciência da Computação*.

Orientador: David Menotti Gomes.

CURITIBA PR

2021

RESUMO

Durante a realização deste trabalho foi implementada uma arquitetura que possibilita realizar com alto nível de acurácia a predição de jogos profissionais do jogo *League of Legends* (*LoL*). A solução proposta contém seis modelos que através de dados consumidos em tempo real, realiza predições e notifica possíveis apostadores. Estes modelos foram desenvolvidos para executar as predições de 0 até 25 minutos de jogo a cada cinco minutos, apresentando uma acurácia consistente (média de 76%) e condizente com os trabalhos semelhantes que utilizam outros jogos como base. Na construção dos modelos foi utilizado o algoritmo de *machine learning* Regressão Logística, pois o mesmo apresentou o melhor resultado para o problema de classificação que este trabalho visa solucionar.

Palavras-chave: *League of Legends*. Aprendizado de máquina. Predição .

ABSTRACT

During the performance of this work, an architecture was implemented that makes it possible to perform the prediction of professional games in the League of Legends (LoL) game with a high level of accuracy. The proposed solution contains six models that, using data consumed in real-time, make predictions and notify potential bettors. These models were developed to perform predictions from 0 to 25 minutes of the game every five minutes, presenting a consistent accuracy (average of 76%) and consistent with similar works that use other games as a basis. In the construction of the models, the machine learning algorithm Logistic Regression was used, as it presented the best result for the classification problem that this work aims to solve.

Keywords: League of Legends. Machine Learning. Prediction.

LISTA DE FIGURAS

2.1	Mapa <i>Summoner's Rift</i> , principal mapa utilizado em <i>League of Legends</i> — Foto: Divulgação/Riot Games.	10
2.2	Mapa de <i>Summoner's Rift</i> com suas posições — Foto: Reprodução (Globo Esporte, 2021).	12
4.1	Arquitetura da solução desenvolvida.	21
4.2	Gráfico com as acurácias (%) dos modelos desenvolvidos.	26
4.3	Gráfico de comparação das acurácias dos modelos por quantidade de dados utilizados.	27

LISTA DE TABELAS

3.1	Tabela informativa com informações sobre os trabalhos anteriores relacionados ao trabalho proposto.	17
-----	---	----

LISTA DE ACRÔNIMOS

DINF	Departamento de Informática
PPGINF	Programa de Pós-Graduação em Informática
UFPR	Universidade Federal do Paraná
LoL	<i>League of Legends</i>
DotA	<i>Defense of the Ancients</i>
API	<i>Application Programming Interface</i>
RTS	<i>Real Time Strategy</i>
MOBA	<i>Multiplayer Online Battle Arena</i>
RPG	<i>Role Playing Game</i>
AUC	<i>Area Under Curve</i>

SUMÁRIO

1	INTRODUÇÃO	8
1.1	MOTIVAÇÃO	8
1.2	OBJETIVO	9
1.3	ORGANIZAÇÃO	9
2	FUNDAMENTAÇÃO TEÓRICA	10
2.1	<i>LEAGUE OF LEGENDS</i>	10
2.1.1	Definição	10
2.1.2	Mapa	10
2.1.3	Posições	11
2.1.4	Campeões	12
2.1.5	Seleção de campeões	12
2.1.6	Lógica do jogo	13
2.1.7	Objetivos	13
2.1.8	Atualizações	14
2.1.9	<i>eSports</i>	14
3	TRABALHOS RELACIONADOS	15
3.1	ANÁLISE PREDITIVA EM MOBA	15
4	DESENVOLVIMENTO	20
4.1	ARQUITETURA	20
4.1.1	Módulo Produtor	21
4.1.2	Módulo Consumidor	22
4.2	BANCO DE DADOS	23
4.3	TRANSFORMAÇÃO DOS DADOS	23
4.4	MODELOS	24
4.5	RESULTADOS	25
4.5.1	Impacto da quantidade de dados no desempenho dos modelos	26
5	CONCLUSÃO	28
5.1	PRÓXIMOS PASSOS/TRABALHOS FUTUROS	28
6	ANEXOS	29
6.1	LISTA DE VARIÁVEIS DOS MODELOS	29
	REFERÊNCIAS	34

1 INTRODUÇÃO

O gênero *Multiplayer Online Battle Arena*, também conhecido como MOBA se popularizou a partir de títulos de sucesso como *League of Legends (LoL)* e *DotA 2* (Povoleri, 2021). Se trata de um estilo de jogo que engloba ação, estratégia e RPG. Os jogos desta modalidade pela grande popularidade detêm as principais modalidades dos esportes eletrônicos (Wikipedia, 2021b).

A origem do MOBA vem da junção de alguns jogos como *Aeon of Strife (AoS)*, um mapa modificado do jogo *Starcraft* inspirado em outro jogo, chamado *Future Cop: LAPD* (Wikipedia, 2021a). Ambos os jogos davam ênfase no estilo de *Real-Time Strategy* (RTS ou estratégia em tempo real) (Povoleri, 2021).

Em 2009, aproveitando a oportunidade onde existia uma modalidade de jogo bastante popular, mas sem títulos dedicados, a *Riot Games* lançou o clássico *League of Legends (LoL)*, que logo se tornou referência no segmento e atualmente é o principal produto do cenário mundial de esportes eletrônicos (Povoleri, 2021).

Podemos definir esportes eletrônicos ou *eSports* como “competições profissionais de *games* que ocorrem em uma plataforma digital, envolvendo dois ou mais competidores (sejam indivíduos ou equipes), em partidas *online* ou presenciais síncronas e montadas para permitir o acompanhamento de uma audiência”. (CBeS, 2019)

O mercado de *games* teve um crescimento de 23,1% entre 2020 e 2021 o que resultou em uma receita de 175,8 bilhões de dólares no total. Esse crescimento muito se deve à pandemia que despertou o interesse pelo seguimento em novos públicos. (Wijman, 2021)

O mercado de esportes eletrônicos assim como o de *streaming* de jogos está em franco crescimento e deve crescer cerca de 70% nos próximos quatro anos movimentando US\$ 3,5 bilhões em 2025 — contra US\$ 2,1 bilhões atualmente, segundo a *Juniper Research*. (Saidat Giwa-Osagie, 2021)

Contextualizando com o mercado de *games* é possível ver que o mercado de esportes eletrônicos tem muito a crescer nos próximos anos tanto em receita quanto em público.

Um dos segmentos que vem movimentando muito dinheiro neste mercado são as apostas esportivas, voltadas aos esportes eletrônicos (Rejane, 2021). Sendo assim, focando no cenário competitivo do *League of Legends*, este trabalho propõe abordagens baseadas em *machine learning* capazes de predizer o vencedor das partidas antes do jogo começar e em tempo real, utilizando dados coletados através de uma *API* pública da *Riot games*, empresa criadora do jogo.

1.1 MOTIVAÇÃO

A motivação neste estudo está em fornecer suporte a um desenvolvimento futuro, onde o objetivo é integrar com plataformas de apostas, e conforme as *odds* (OddsShark, 2018) (estimativa

de retorno por valor apostado, definido pela casa de aposta de acordo com alguns fatores, como, por exemplo: favoritismo, momento em jogo, quantia aportada por outros apostadores, etc) correntes de cada jogo realizar entradas e saídas automáticas, tendo um retorno financeiro positivo.

1.2 OBJETIVO

O objetivo principal deste trabalho consiste em desenvolver uma solução capaz de prever com um alto nível de assertividade os vencedores de partidas profissionais do jogo, facilitando o processo de tomada de decisão de apostadores desta categoria de mercado.

Os objetivos específicos são:

- Além disso, será desenvolvida a arquitetura responsável por notificar observadores interessados nas previsões dos modelos desenvolvidos para jogos profissionais em andamento, possibilitando entradas manuais em sites de apostas.
- Após a conclusão do desenvolvimento, a intenção é metrificar a performance dos modelos propostos em ambiente produtivo com jogos e apostas reais.

1.3 ORGANIZAÇÃO

O documento está organizado da seguinte forma. No Capítulo 2, é apresentado um resumo referente ao jogo, onde diversos conceitos e termos são apresentados ao leitor para que a compreensão do trabalho seja facilitada.

Em seguida, no Capítulo 3, é apresentada uma descrição dos trabalhos relacionados. A maioria dos trabalhos relacionados desenvolvidos anteriormente abordam outros jogos ou não limitam o domínio estudado ao nível profissional. O Capítulo 4 aborda o desenvolvimento do projeto, onde são descritos diversos pontos importantes para entendimento da solução proposta pelo autor.

Temos no Capítulo 5 a conclusão do autor a respeito da solução proposta.

Por fim temos o Capítulo 6 onde são apresentadas como anexo a lista de variáveis utilizadas com a descrição do que significa cada variável.

2 FUNDAMENTAÇÃO TEÓRICA

2.1 LEAGUE OF LEGENDS

2.1.1 Definição

Segundo a própria *Riot Games* (Games, 2021), “*League of Legends* é um jogo de estratégia em que duas equipes de cinco poderosos campeões se enfrentam para destruir a base uma da outra”. Destruir a base do time inimigo se trata de destruir o objetivo principal do outro time, conhecido como *Nexus* (coração da base de uma equipe, ao destruir o *Nexus* da equipe inimiga sua equipe vence), o que será explicado na Seção 2.1.7.

2.1.2 Mapa

O mapa do jogo é conhecido como *Summoner's Rift*, é espelhado de forma diagonal e conta com uma região denominada como Selva e três rotas que ligam a base de uma equipe a outra. Cada rota possui três torres e um inibidor e são interligadas por caminhos que se cruzam na selva. As rotas são conhecidas como: rota superior (*top*), rota do meio (*mid*) e rota inferior (*bot*).

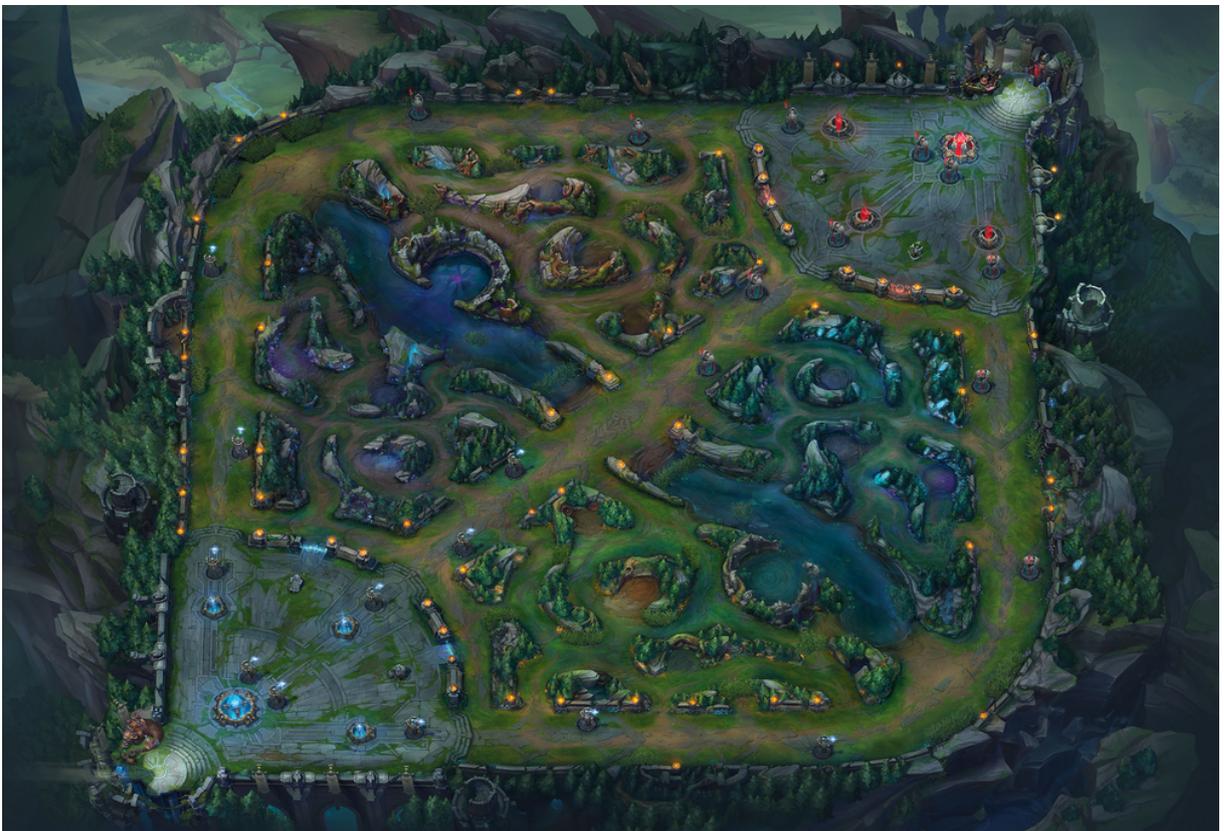


Figura 2.1: Mapa *Summoner's Rift*, principal mapa utilizado em *League of Legends* — Foto: Divulgação/Riot Games

2.1.3 Posições

Os times são compostos de 5 jogadores, onde cada jogador ocupa uma posição distinta no mapa. Cada posição, contém uma responsabilidade distinta no jogo do time e pode variar de acordo com a estratégia utilizada pelo time. As responsabilidades descritas a seguir dizem respeito ao que acontece geralmente nas partidas profissionais e em jogos de alto nível.

- **Topo:** Geralmente é a posição mais isolada do jogo, ficando separada das principais ações que acontecem durante a partida. Nesta rota é bem comum que as equipes escolham campeões com mais vida e resistência, que conseqüentemente suportam mais dano e fiquem na linha de frente para os campeões carregadores escolhidos pelo time darem dano nos campeões inimigos nas lutas.
- **Caçador/Selva:** Posição que é responsável por fazer o jogo girar, funcionando como uma engrenagem para o time. De cada lado do mapa existem duas selvas, a superior e a inferior. O caçador do time consegue transitar entre ambas as selvas do seu time e além disso invadir a selva do time adversário. No jogo, o caçador auxilia os aliados seja através de lutas, pressionando algum jogador de rota do time inimigo, fazendo cobertura para possíveis emboscadas do caçador inimigo, surpreendendo o inimigo na rota ou auxiliando no controle de visão de posições estratégicas.
- **Meio:** Posição mais flexível do jogo, o que permite aos jogadores jogar com campeões de diferentes funções, desde assassinos até mesmo campeões utilitários. Por estar em uma posição intermediária no mapa tem acesso rápido a selva e às rotas mais afastadas podendo adquirir vantagens para o time ao realizar esse deslocamento.
- **Atirador:** Uma das posições que costuma jogar na rota inferior, geralmente joga com campeões responsáveis por deferir maior parte do dano da equipe, logo tem a preferência na acumulação de recursos permitindo o mesmo a comprar mais itens, ficando mais forte na partida. Campeões escolhidos para essa posição podem variar de acordo com o meta, entretanto na grande maioria das partidas são escolhidos carregadores de dano físico à distância.
- **Suporte:** Como o próprio nome da posição diz, é responsável por auxiliar o time, principalmente o Atirador na rota inferior. Os campeões escolhidos para esta posição geralmente são os campeões utilitários.



Figura 2.2: Mapa de *Summoner's Rift* com suas posições — Foto: Reprodução (Globo Esporte, 2021)

2.1.4 Campeões

O jogo contém mais de 150 campeões e cada campeão apresenta uma função específica com habilidades e maneiras de se jogar diferentes. Os campeões se dividem nas seguintes classes.

- Assassinos
- Atiradores
- Lutadores
- Magos
- Suportes
- Tanques

2.1.5 Seleção de campeões

A seleção de campeões se trata de fase pré-jogo, onde cada equipe precisa escolher os campeões que serão utilizados na partida. Como cada campeão possui características únicas é muito importante que as equipes combinem personagens que apresentam boa sinergia contra os personagens da equipe adversária. Sendo assim temos a fase de *Draft* (como é chamado

esse processo de escolha). Resumidamente as equipes intercalam entre duas fase de escolhas e banimentos de forma alternada, resultando em cinco remoções de campeões de cada lado e cinco escolhas.

2.1.6 Lógica do jogo

Os campeões escolhidos de cada time nascem nas suas respectivas bases e rotacionam pelas rotas do mapa buscando a conquista dos principais objetivos. Os campeões ficam mais fortes conforme vão adquirindo recursos dentro do jogo (*gold*) e comprando itens além do ganho de experiência que permite com que haja evolução de nível e consequentemente aumento das características básicas de cada personagem.

Esse acúmulo de recursos vem de diversas fontes como, abate de tropas (*minions* que nascem no *Nexus* de cada time e formam pequenas tropas auxiliando os personagens principais), abates de campeões, conquistas de objetivos globais, além de uma quantia de ouro recebida de acordo com o tempo da partida.

Os campeões se enfrentam para conquistas os objetivos onde o maior deles se trata do *Nexus* inimigo, o que resulta na vitória do time que consegue este feito.

2.1.7 Objetivos

Como dito anteriormente, o jogo se trata basicamente de destruir o *Nexus* do time inimigo. Entretanto, para conseguir isso é necessário a conquista de outros objetivos, são eles: Torre, Inibidor.

Uma torre pode ser definida como uma fortificação que bloqueia o avanço do time inimigo ao *Nexus* da outra equipe. Elas causam dano e dão visão ao seu arredor.

Outro objetivo essencial para a finalização de um jogo é o Inibidor. Um Inibidor se trata de uma estrutura que bloqueia a formação de super tropas inimigas na rota em que o inibidor está localizado.

Além desses objetivos essenciais, o jogo possui outros três objetivos globais, não essenciais, mas geralmente conquistados pelas equipes vencedoras dos confrontos, são eles: Dragões, Arauto e Barão.

Os dragões são representações de elementos, nascem de forma aleatória a cada 5 minutos a partir do quinto minuto de jogo. Cada Dragão tem um efeito diferente(Esporte, 2021):

- **Dragão Infernal:** Concede aumento percentual de dano.
- **Dragão das Nuvens:** Concede redução do tempo de recarga para habilidade especial.
- **Dragão do Oceano:** Concede regeneração de vida perdida (percentual) a cada segundo.
- **Dragão da Montanha:** Concede aumento percentual de armadura e resistência mágica.
- **Dragão Ancião:** Executa adversários com vida baixa após entrar em confronto.

O Arauto se trata de um objetivo que nasce até duas vezes no jogo, na mesma posição do barão e quando conquistado pode ser invocado pelo jogador que o capturou pelos próximos 4 minutos. Quando invocado desperta o *Arauto do Vale* que desperta e causa dano as torres do time inimigo na rota em que é despertado (Esporte, 2021).

O Barão, conhecido como *Barão de Na'Shor* surge a partir de 20 minutos de jogo no lugar do Arauto e é considerado o principal objetivo (tirando o *Nexus*) do *League of Legends*. Quando conquistado concede diversas vantagens aos campeões da equipe que derrotou o Barão por 210 segundos(Tartaglia, 2019).

2.1.8 Atualizações

O *League of Legends* é um jogo bastante dinâmico e versátil(Esporte, 2021) e grande parte disso está nas atualizações quinzenais que o jogo recebe, onde acontece o balanceamento dos campeões, posições, itens e runas que estão fora da curva, tanto para o lado positivo quanto para o lado negativo.

2.1.9 eSports

A *Riot Games* adota um sistema regionalizado para o competitivo de *LoL*, sendo assim, regiões de maior expressão possuem campeonatos próprios e campeonatos internacionais acontecem duas vezes por ano. Os campeonatos regionais acontecem em duas temporadas, conhecidas como *split* que são separadas pelas competições internacionais existentes, o *Mid-Season Invitational (MSI)* que acontece no meio do ano e o *Worlds* que acontece no final do ano após o segundo *split*. Os participantes destes campeonatos são geralmente os melhores times das competições regionais.

3 TRABALHOS RELACIONADOS

3.1 ANÁLISE PREDITIVA EM MOBA

Nesta seção apresentamos uma revisão da literatura a respeito de trabalhos relacionados ao tema que foi proposto. Não foi encontrado trabalhos onde o objetivo era utilizar dados históricos para predição de acontecimentos em jogos profissionais do *League of Legends*. Sendo assim, foi utilizado como referência outros trabalhos que possivelmente poderiam ajudar no desenvolvimento do tema para que o objetivo do trabalho em questão fosse alcançado com os melhores resultados possíveis.

Prever vencedores de partidas profissionais de MOBA (*Multiplayer Online Battle Arena*) com acurácia relevante é algo possível e alcançável (Hodge et al., 2019). Dentro desta modalidade de jogos eletrônicos, temos dois títulos que apresentam uma variedade maior de trabalhos relacionados, são eles DotA 2 e *League of Legends*. O jogo DotA 2 é um MOBA muito similar ao *League of Legends* pois ambos são descendentes do Defense of the Ancients (DotA, uma modificação em um mapa desenvolvido para o jogo *Warcraft III: The Frozen Throne*). O jogo em questão é desenvolvido pela produtora *Valve Corporation* sendo um dos principais carros chefes da produtora.

Os trabalhos envolvendo predição de vencedores em jogos de DotA 2 utilizavam modelos simples e desconsideravam as interações entre os personagens na fase de escolhas (Semenov et al., 2017). O artigo em questão realizou uma comparação de performance dos algoritmos de *machine learning* mais utilizados, e além de introduzir a utilização de *Factorization Machines* para este tipo de problema foi o primeiro trabalho relacionado ao tema a mostrar que a acurácia dos modelos varia quando levado em conta o nível de habilidade dos jogadores. Segundo os autores, outro ponto que ficava a desejar nos trabalhos anteriores estava nas datas dos dados coletados para criação dos *datasets*, uma vez que como este estilo de jogo recebe atualizações constantes caso haja uma mudança brusca no jogo os dados anteriores deixam de ter tanta relevância para futuras predições, fato este que não era levado em conta nos trabalhos relacionados. Considerando somente jogadores de alto nível, os autores conseguiram uma AUC (*Area Under the ROC Curve*) de 66% utilizando o modelo de *Factorization Machines*.

Em 2019, utilizando dados em tempo real de partidas de Dota 2 pesquisadores conseguiram prever com uma acurácia de 85% o vencedor de um jogo (Hodge et al., 2019). O estudo foi intitulado pelos próprios autores como o primeiro caso de estudo de predição em tempo real em jogos profissionais de eSports e além dos algoritmos de *machine learning* padrão para o desenvolvimento dos modelos, foi utilizado *feature engineering* e otimização. Durante a realização do trabalho, os autores chegaram a validar os modelos implementados no maior campeonato internacional de DotA 2 do ano em questão, a *ESL One Hamburg 2017*, analisando

o jogo entre 5 e 30 minutos, o método obteve uma acurácia na predição do vencedor do confronto variando entre 70% a 90%.

Tanto *League of Legends* como DotA apresentam uma alta similaridade nos aspectos gerais, sendo possível utilizar os trabalhos relacionados de um como referência para o outro. Os trabalhos anteriores relacionados a predição em MOBAs adotaram uma variedade de características representando diferentes aspectos das partidas que foram treinados em uma variedade de algoritmos de aprendizado de máquina. Geralmente os vetores de características utilizados para a predição contêm dados que podem ser divididos em três faixas, são elas:

- Características pré-jogo: São características estatísticas do jogo ou dos jogadores com base nos últimos jogos, desempenho no campeonato, histórico contra o adversário em questão.
- Características em-jogo (tempo real): São características extraídas durante a realização do jogo. Em suma, são características tanto do jogo quanto dos jogadores voltadas ao contexto do jogo em questão. Como por exemplo, tempo de jogo até o momento, número de torres destruídas, dragões/barões executados e etc.
- Características pós-jogo: São características estatísticas sumarizadas após o final do jogo e utilizadas para retreinar o modelo na tentativa de aumentar a acurácia da predição de jogos futuros.

Usando como base apenas o jogo *League of Legends*, Tian Wang em sua dissertação de mestrado apresentou um trabalho discutindo as *features* com maior potencial para interferir na predição de jogos competitivos de LoL (Wang, 2018). O trabalho utilizou tanto dados históricos (características pós-jogo) como informações de *champion select* (momento em que é realizado as escolhas e banimentos de personagens de cada time) e informações extraídas durante o jogo. Segundo Tian, "se um time tem mais torres destruídas que o time inimigo, conseguiu o primeiro abate, a primeira torre e o primeiro barão, esse time tem uma chance mais alta de vencer o jogo" (Wang, 2018), isto acontece pois segundo o trabalho essas quatro variáveis mostraram uma influência maior que as outras variáveis em determinar o resultado de uma partida.

Os demais trabalhos relacionados em grande maioria fizeram uso de dados estatísticos/*features* do jogo em questão computados após os jogos anteriores de cada confronto (Almeida et al., 2017; Semenov et al., 2017; Kim et al., 2020; Wang, 2016; Wang et al., 2018). Fazendo uma breve comparação com os trabalhos que utilizaram dados extraídos em *real time* (Wang, 2018; Hodge et al., 2019; Yang et al., 2016, 2014; Hodge et al., 2017; Yang et al., 2020; Makarov et al., 2018), é possível concluir que ter essas características extraídas durante o andamento do jogo ajudam a construir modelos com acurácia superior aos modelos que não fazem utilização delas.

Relacionado a modelos de *machine learning*, um dos trabalhos relacionados ao tema proposto que utilizou como base o *League of Legends* foi único entre os trabalhos relacionados a

utilizar o Perceptron como modelo para predição de vencedores em partidas de eSports (Kim et al., 2020). Os demais, em grande maioria utilizaram *Logistic Regression*, *Neural Network*, *Decision Tree*, *Factorization Machines*, *Support Vector Machines* e *Random Forest* (Almeida et al., 2017; Semenov et al., 2017; Kim et al., 2020; Wang, 2016; Wang et al., 2018; Wang, 2018; Hodge et al., 2019; Yang et al., 2016, 2014; Hodge et al., 2017; Yang et al., 2020; Makarov et al., 2018).

Tabela 3.1: Tabela informativa com informações sobre os trabalhos anteriores relacionados ao trabalho proposto.

Título	Objetivo	Base de dados	Modelos
<i>Predictive Analysis on eSports Games: A Case Study on League of Legends (LoL) eSports Tournaments</i>	Discutir como diferentes características influenciam os resultados de jogos competitivos de <i>league of legends</i>	Dois <i>datasets</i> com pouco mais de 1.000 registros coletados da API de dados fornecida pela <i>Riot games</i> , proprietária do título <i>League of Legends</i> : O primeiro com 200 jogos, com <i>features</i> extraídas durante o jogo. O segundo com 855 jogos com informações pré-jogo.	Regressão Logística e Árvore de decisão
<i>Win Prediction in Multi-Player Esports: Live Professional Match Prediction</i>	Predição de vencedores em partidas profissionais do jogo DotA 2	<i>Dataset</i> com 5.700 jogos sendo 24% partidas profissionais e 76% partidas amadoras de <i>ranking</i> alto. Dados coletados através de <i>replay</i> obtidos pelo <i>OpenDota</i>	<i>Random Forest</i> e Regressão Logística
<i>Real-time eSports Match Result Prediction</i>	Predição de vencedores em partidas do jogo DotA 2	<i>Dataset</i> com 78.362 jogos alto nível coletados da API oficial do Dota 2.	Regressão Logística e Rede Neural
Continua na próxima página			

Tabela 3.1 – continuação da página anterior

Título	Objetivo	Base de dados	Modelos
<i>Identifying Patterns in Combat that are Predictive of Success in MOBA Games</i>	Identificar padrões que identificassem com sucesso os vencedores de uma partida no jogo DotA 2	<i>Dataset</i> com 407 logs de jogos competitivos de DotA 2, obtidos através da <i>Go-suGamers</i> e da <i>getdotastats.com</i> que são comunidades online para jogadores de DotA.	Árvores de Decisão
<i>Win Prediction in Esports: Mixed-Rank Match Prediction in Multi-player Online Battle Arena Games</i>	Predição de vencedores em partidas do jogo DotA 2	<i>Dataset</i> com 1.933 jogos obtidos através do site da <i>Valve</i> , proprietária do DotA 2.	Regressão logística e <i>Random Forest</i>
<i>Prediction of winners in MOBA games</i>	Predição de vencedores em partidas do jogo DotA 2	<i>Dataset</i> com 123.326 partidas do jogo DotA 2.	<i>Naive Bayes</i> , KNN e Árvore de Decisão
<i>Predicting Events in MOBA Games: Dataset, Attribution, and Evaluation</i>	Predição de eventos no jogo <i>Honor of Kings</i>	<i>Dataset</i> com 50.279 partidas de <i>Honor of Kings</i> com informações de <i>frames</i> da partida observada.	LSTM e <i>Transformer</i>
<i>Predicting Winning Team and Probabilistic Ratings in “Dota 2” and “Counter-Strike: Global Offensive” Video Games</i>	Predição de vencedores em partidas dos jogos <i>Counter Strike: Global Offensive</i> e DotA 2	-	Regressão Logística, <i>Support Vector Machines</i> , <i>Gradient Boosting</i> , e <i>Random Forest</i> .
<i>Performance of Machine Learning Algorithms in Predicting Game Outcome from Drafts in Dota 2</i>	Metrificar a performance de algoritmos de <i>machine learning</i> ao prever partidas de DotA 2 após a realização do <i>draft</i>	<i>Dataset</i> 5.071.858 partidas com dados coletados da API da <i>Steam</i> .	Naive Bayes, Regressão Logística, <i>Gradient Boosted</i> , <i>Decision Trees</i> e <i>Factorization Machines</i>
Continua na próxima página			

Tabela 3.1 – continuação da página anterior

Título	Objetivo	Base de dados	Modelos
<i>A Confidence-Calibrated MOBA Game Winner Predictor</i>	Predição de vencedor em partidas de <i>League of Legends</i>	<i>Dataset</i> com 93.875 partidas de <i>League of Legends</i> .	<i>Perceptron</i>
<i>Predicting Multiplayer Online Battle Arena (MOBA) Game Outcome Based on Hero Draft Data</i>	Predição de vencedor em partidas de DotA 2	<i>Dataset</i> 5.071.858 partidas com dados coletados da API da <i>Steam</i> .	Regressão Logística, Rede Neural e SGD
<i>Outcome prediction of DOTA2 using machine learning methods</i>	Predição de vencedor em partidas de DotA 2	<i>Dataset</i> contendo 38.629 partidas das ligas chinesas de alto nível coletados através da API da <i>Valve</i> , proprietária do título DotA 2.	Regressão Logística, <i>Random Forest</i> , <i>Support Vector Machines</i>

4 DESENVOLVIMENTO

O desenvolvimento do trabalho em questão foi iniciado visando prever resultados de jogos profissionais do jogo *League of Legends*. Para tal, era necessário realizar a construção de uma arquitetura robusta o suficiente para ser possível a obtenção de dados históricos, assim como os dados em tempo real.

Após a leitura dos trabalhos anteriores relacionados ao tema era clara a necessidade da utilização de algum algoritmo de *machine learning* para a realização da predição do resultado. O desafio começou com a obtenção dos dados necessários para o desenvolvimento dos modelos, visto que não foi encontrada uma fonte confiável dos dados demandados.

Logo, para unir a necessidade da obtenção de dados históricos com a necessidade de dados em tempo real foi necessário desenvolver um *API Client* para consumir diretamente os *endpoints* que o portal de *eSports* da própria *Riot Games*¹ consome, satisfazendo assim todas as necessidades do projeto.

Com o desenvolvimento deste *client* foi possível arquitetar a solução para solucionar o problema proposto e construir uma solução de ponta a ponta, onde dado eventos acontecendo em tempo real no nível profissional do jogo a predição do vencedor do confronto é realizada e os interessados são notificados.

4.1 ARQUITETURA

A solução para o problema foi arquitetada para funcionar de forma incremental trazendo mais consistência e resiliência para os modelos desenvolvidos com o decorrer do tempo. Isso se dá, pois, teremos mais observações e sendo assim é possível realizar o treinamento com mais dados históricos.

A arquitetura da solução contém dois módulos principais com múltiplas funções, chamados módulo produtor e módulo consumidor. Além disso, um banco de dados utilizando o gerenciador de banco de dados *PostgresQL*² foi construído e populado com as informações históricas dos jogos profissionais do jogo em questão.

Podemos observar na figura 4.1 um diagrama que mostra o fluxo de como a comunicação entre esses módulos acontece:

¹<https://lolesports.com/>

²<https://www.postgresql.org/>

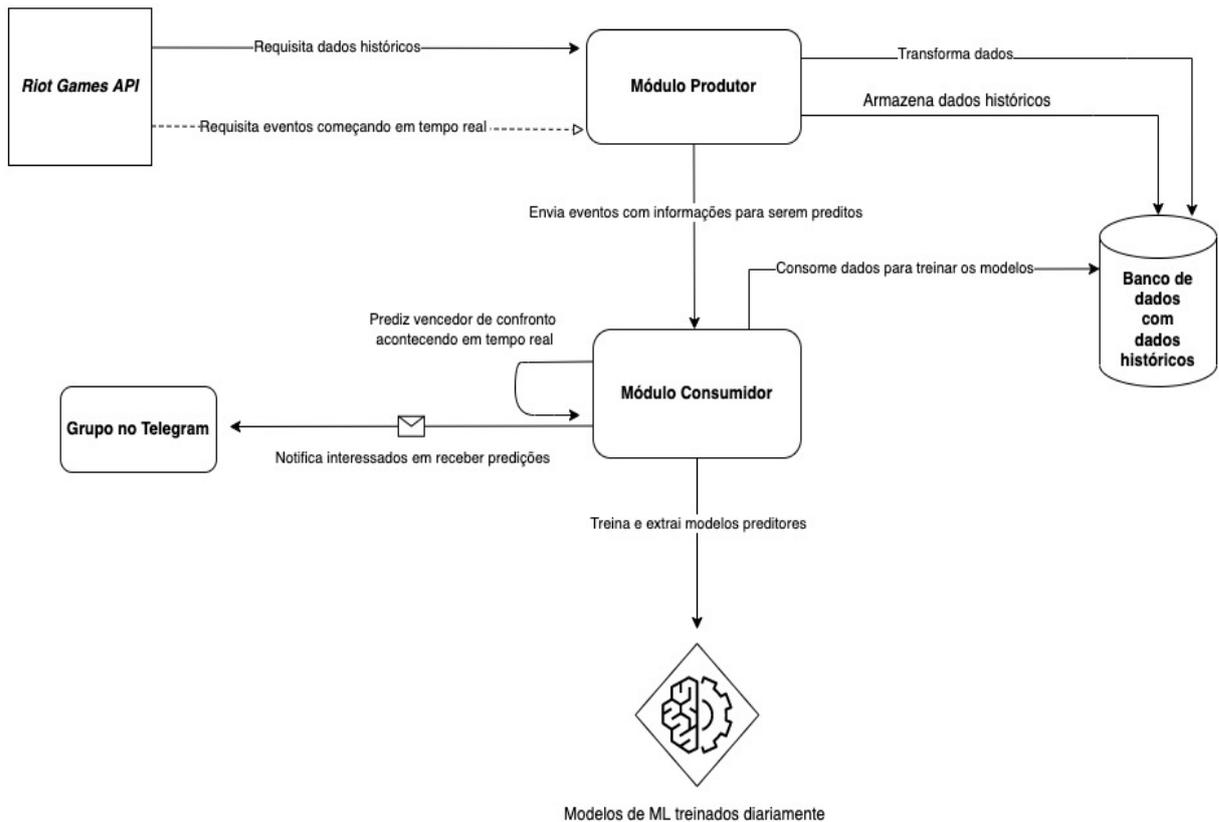


Figura 4.1: Arquitetura da solução desenvolvida.

4.1.1 Módulo Produtor

O módulo produtor é uma aplicação monolítica escrita em Go³. É o módulo principal da solução proposta, haja visto que este módulo tem a responsabilidade de subsidiar a obtenção dos dados e fazer o monitoramento para realizar o *tracking* de novos eventos a serem preditos.

Às três principais funções deste módulo, são:

- Sincronização do banco de dados. A rotina de sincronização do banco de dados foi desenvolvida para possibilitar a execução diária, de forma incremental, persistindo os dados dos eventos que não estavam sincronizados com a base de dados no estado atual. Os dados utilizados são obtidos através da *API* da *Riot Games*, usando o *API client* também implementado neste módulo.
- Engenharia de característica. O módulo produtor é responsável por executar uma rotina para realização de *feature engineering*, fazendo transformações de dados e gerando novas variáveis não disponíveis na *API* da *Riot Games*, com o intuito de aumentar as variáveis disponíveis para o treinamento dos modelos e possivelmente gerando uma maior acurácia.

³<https://go.dev/>

- *Tracking* de eventos em *real time*. Este módulo carrega a responsabilidade através de um *API pooling* que ao identificar um novo evento acontecendo em tempo real, dispara processos agendados no futuro com requisições ao outro módulo da solução proposta. O conteúdo nessas requisições agendadas são importantes para o módulo consumidor, visto que neste conteúdo estão presentes a identificação do evento, variáveis construídas pela rotina do item anterior, além de qual modelo que deve ser utilizado para a realização da predição.

4.1.2 Módulo Consumidor

É uma aplicação monolítica escrita em Python⁴. Este módulo recebe as requisições com pedidos de predição de eventos que estão acontecendo em tempo real.

As funções desse módulo são expostas por *endpoints* de uma *API Rest*, cada *endpoint* pode ser descrito da seguinte forma:

- O primeiro é um *endpoint* responsável por manipular e tratar os dados extraídos da base de dados populada pelo módulo produtor e, além disso, realizar o treinamento dos seis modelos desenvolvidos e extraí-los para cada momento de jogo pré-determinado.

Conforme dito anteriormente, foram definidos momentos específicos para a coleta dos dados de jogo, começando após a fase de *champion select* (Capítulo 2) até os 25 minutos de jogo, a cada cinco minutos.

Sendo assim, um modelo específico para cada momento de coleta de dados foi desenvolvido visando especializar a predição, realizando o treino dos modelos com dados do momento exato em que está sendo realizada a predição na partida atual.

Estes modelos são retreinados diariamente, considerando que a persistência de forma incremental de novos eventos pode disponibilizar mais dados para treinamento.

- O segundo é o *endpoint* utilizado na comunicação com o módulo produtor. Este *endpoint* recebe os dados de um novo evento mapeado a cada cinco minutos a partir do seu início. Nestes dados estão contidas variáveis históricas e variáveis geradas de ambas as equipes, além do momento de jogo á que as *features* se referem para indicar qual modelo será utilizado na predição.

Com estas variáveis recebidas na requisição o módulo realiza requisições a *API da Riot Games* para obter as variáveis da partida ao vivo, carrega o modelo a ser utilizado e realiza a predição.

Com o vencedor do confronto predito é emitida uma notificação em um canal de comunicação. Essas notificações são enviadas em um grupo de um serviço de mensagens instantâneas baseado na nuvem, o Telegram ⁵.

⁴<https://www.python.org/>

⁵<https://pt.wikipedia.org/wiki/Telegram>

4.2 BANCO DE DADOS

Foi construído um banco de dados com todas as informações passíveis de serem extraídas da API pública utilizada, onde foram sumarizados até o momento dados de 9572 jogos profissionais de todas as ligas do mundo e campeonatos internacionais.

Além das informações dos jogos, foram coletadas estatísticas das ligas, equipes e jogadores nos momentos oportunos onde os modelos foram treinados.

Um dos desafios para o desenvolvimento da solução proposta estava justamente em construir o banco de dados com os dados históricos de partidas profissionais do jogo *League of Legends*. Com a construção deste banco de dados se tornou possível montar a base de dados específica para cada modelo a ser treinado, utilizando os dados dos times e dos jogadores considerando o tempo da partida.

Alguns problemas relacionados aos dados foram encontrados, dado que a fonte de dados utilizada no trabalho (API da *Riot Games*) falha ao fornecer informações sobre momentos específicos de algumas partidas, fazendo com que os dados destas partidas sejam descartados da análise do trabalho.

As competições dos esportes eletrônicos segue um calendário semelhante aos esportes tradicionais, logo a sincronização do banco de dados com partidas recentes será automatizada e executada diariamente incrementando a quantidade de jogos profissionais disponíveis para treinamento dos modelos.

4.3 TRANSFORMAÇÃO DOS DADOS

Conforme mencionado anteriormente, a engenharia de característica realizada no módulo produtor é efetuada por um *worker* construído neste módulo. A transformação de dados realizada é executada diariamente, sempre após a rotina de sincronização dos dados.

O intuito é gerar novas variáveis com alto grau de predizibilidade, não disponibilizadas pela fonte de dados do projeto, para todos os dados presentes no banco de dados populado.

Neste processo, a média dos dados históricos dos 3 e 5 últimos confrontos de cada equipe são calculadas. A intenção é gerar variáveis que infiram um momento de cada equipe, antes do confronto entre ambas.

Para esta análise são considerados os 15 e 25 minutos de uma partida como pontos-chaves. Sendo assim, é possível identificar como cada equipe vem lidando estrategicamente com o jogo recentemente, por exemplo: uma equipe do confronto ter na sua forma recente uma média superior ao adversário na obtenção de recursos aos 15 minutos de jogo, indica que possivelmente esta mesma equipe vem sendo mais agressiva no começo das partidas.

No total foram geradas 48 novas variáveis neste processo para serem incluídas nos modelos. A lista completa dessas variáveis pode ser encontrada no Capítulo 6.

4.4 MODELOS

Foram construídos seis modelos distintos, onde cada modelo é responsável por prever o vencedor de um confronto dado o tempo de jogo. O primeiro modelo corresponde a predição do jogo logo após a *champion select* (definido no Capítulo 2), onde os campeões de cada time foram selecionados e a única informação passível de ser utilizada é quais são os campeões de ambos os times e as informações estatísticas históricas geradas pelo processo de transformação de dados. Os demais modelos correspondem a predição do vencedor da partida, com ela ainda em andamento. São mais cinco modelos, considerando dados coletados a cada cinco minutos de jogo.

Esses outros modelos consideram informações do próprio jogo coletadas em tempo real e algumas informações históricas. Essas *features* são relacionadas tanto aos times quanto aos jogadores de cada time.

As variáveis históricas são algumas como, por exemplo: número de vitórias e número de derrotas de cada time no campeonato, além das *features* relacionadas a cada equipe obtidas na transformação dos dados.

As *features* em tempo real são condizentes tanto ao time quanto aos jogadores e são coletadas no momento indicado pelo modelo, ou seja, para o modelo de 5 minutos de jogo os dados são coletados quando o jogo chega neste momento, por exemplo.

Algumas dessas variáveis são: ouro recebido por jogador, percentual de dano aplicado em relação ao time, quantidade de *minions* eliminados, *wards* colocadas e destruídas, etc. A lista com todas as variáveis utilizadas pode ser encontrada no Capítulo 6.

É importante mencionar que um jogo não tem um tempo limite para acabar, logo o jogo pode ter sua duração superior aos 25 minutos, tempo máximo onde modelos foram desenvolvidos. O limite de desenvolvimento de modelos apenas até 25 minutos está no fato de que, considerando a média dos jogos de jogadores do nível desafiante (*ranking* mais elevado do jogo) temos que os jogos acabam na maioria antes dos 30 minutos de jogo (LeagueOfGraphs, 2021).

O desenvolvimento destes modelos foram pautados em extrair os dados do banco de dados, tratar estes dados e agregar em uma observação as variáveis dos dez jogadores que estão presentes na partida, cinco em cada equipe e treinar os modelos utilizando algum algoritmo de *machine learning*.

Durante o desenvolvimento dos modelos vários algoritmos de *machine learning* foram testados, com ênfase nos que estavam presentes na literatura resolvendo problemas semelhantes. Dentre estes algoritmos, temos os seguintes: Regressão Logística, *Naive Bayes*, KNN, SVM, Árvore de Decisão e *Random Forest*.

Esses algoritmos foram estudados e utilizados durante a fase de desenvolvimento dos modelos, por se tratarem de algoritmos que conhecidamente lidam bem com classificação, logo o problema que como objetivo deste trabalho estamos se propondo a resolver.

O algoritmo que apresentou o melhor resultado durante a realização do treinamento/teste dos modelos foi o algoritmo de Regressão Logística. Para todos os momentos de jogo esse algoritmo apresentou uma maior acurácia na classificação dos dados.

Através da utilização deste algoritmo, também foi constatado uma menor variação entre as acurácias do primeiro e último modelo (momento inicial do jogo e 25 minutos de jogo), o que era desejado desde o início do desenvolvimento, haja visto que a intenção é prever com uma acurácia relevante a partida com todos os modelos propostos.

Os dados utilizados durante a fase de treinamento e teste foram obtidos e utilizados ordenadamente por data/horário de realização, nas proporções de 80% e 20% respectivamente. Logo, para qualquer quantidade de dados utilizada durante o treinamento, 80% dos dados mais antigos era utilizado para treinamento e o restante utilizado nos testes.

Diariamente os modelos desenvolvidos são retreinados com a intenção de utilizar novas entradas do banco de dados após o processo de sincronização do mesmo.

4.5 RESULTADOS

Ao utilizar todos os dados históricos disponíveis para treinamento/teste foi possível chegar a um resultado satisfatório para o trabalho proposto. A menor acurácia obtida foi com o modelo que prevê o vencedor de um confronto aos 5 minutos de jogo. No caso este modelo apresentou a acurácia de 72.6% na base de teste (os 20% dos últimos jogos presentes em nossa base).

Por outro lado, a maior acurácia, conforme o esperado, foi obtida com o último modelo (25 minutos de jogo), onde geralmente temos os jogos mais definidos o que possibilita a realização de previsões mais assertivas.

É importante pontuar a acurácia do modelo desenvolvido previamente ao começo da partida. Este modelo apresentou uma acurácia de 74.7%, o que chama bastante atenção, pois para o treinamento deste modelo em questão só temos variáveis históricas disponíveis e somente com estas variáveis foi possível obter um resultado bastante significativo.

Os resultados podem ser observados no gráfico a seguir, onde temos o percentual de acurácia obtida por modelo:

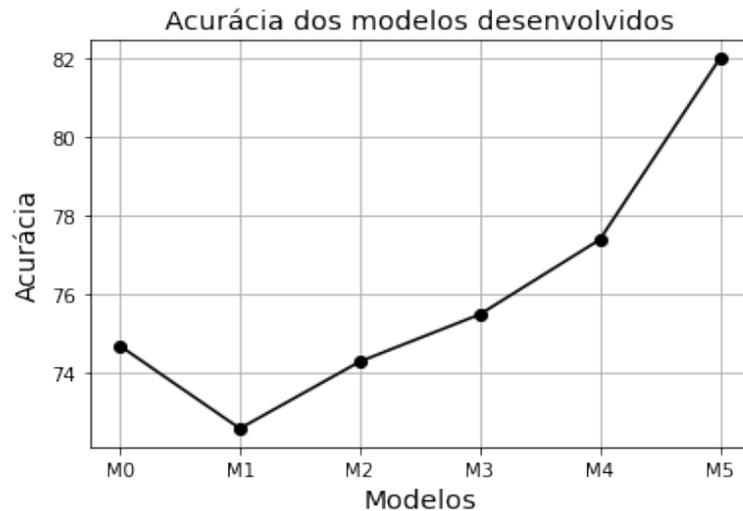


Figura 4.2: Gráfico com as acurácias (%) dos modelos desenvolvidos.

Os resultados obtidos são interessantes por se tratar de um trabalho inicial e um domínio ainda pouco explorado. A previsibilidade de jogos profissionais é menor que quando comparada com a de jogos de jogadores casuais visto que a discrepância de habilidades individuais é menor e detalhes às vezes não mapeados em variáveis estatísticas podem ser decisivos para a determinação do vencedor de um confronto.

É importante mencionar que com a alta acurácia dos modelos ao classificar os vencedores dos eventos, a motivação do trabalho proposto será alcançada com sucesso, haja visto que fornecer informações em média 76% corretas a pessoas que utilizam serviços das plataformas de apostas nessa categoria de mercado seria de grande ajuda na tomada de decisão de quando realizar ou não uma aposta.

O resultado tem uma relação direta com a quantidade de dados disponíveis para treinamento e é possível observar com o estudo a seguir que com o aumento destes dados a acurácia de cada modelo tende a se manter elevada e com pouca variação de um modelo para outro.

4.5.1 Impacto da quantidade de dados no desempenho dos modelos

Foi realizado um estudo para estimar com base no número de observações a eficácia dos modelos, pois como teremos com o decorrer do tempo mais dados históricos, existe o questionamento de que se o modelo irá prever o resultado dos jogos atuais com mais eficiência caso o mesmo seja treinado com a totalidade dos dados ou com apenas as observações mais recentes, dado que as atualizações mais recentes do jogo trazem peculiaridades que ainda não existiam nos jogos passados como, por exemplo: novos campeões, novas estratégias e variação da importância entre os objetivos do jogo.

Sendo assim, os dados disponíveis foram divididos por data de realização em 5 grupos. Cada grupo consistia em um período de tempo distinto, e o conjunto de dados de cada grupo possui um certo número de observações e de variáveis.

Os grupos de dados podem ser definidos como: a totalidade dos dados sem restrição de tempo, os dados dos dois últimos anos, os dados do ano atual, os dados dos últimos seis meses e os dados somente dos últimos três meses.

O resultado deste estudo pode ser observado no gráfico da Figura 4.3.

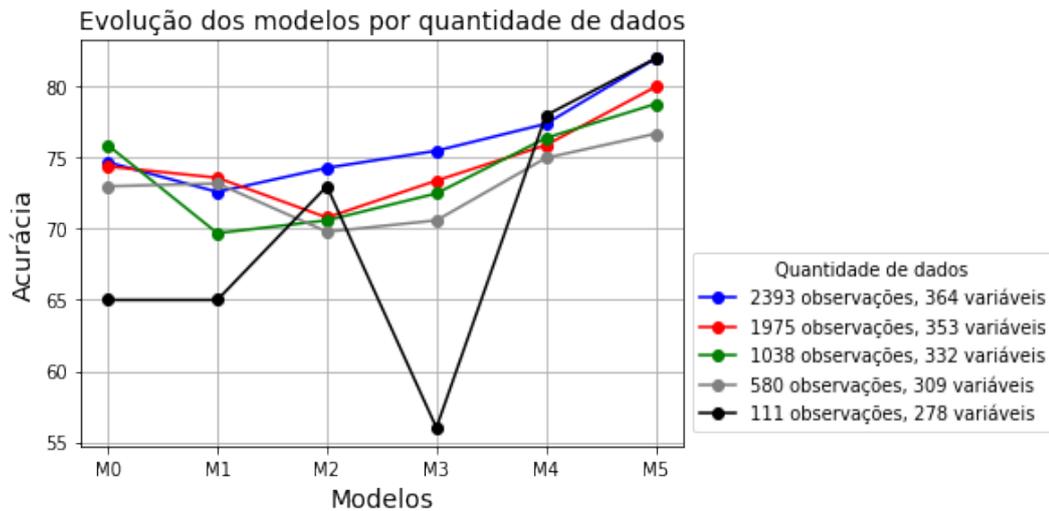


Figura 4.3: Gráfico de comparação das acurácias dos modelos por quantidade de dados utilizados.

O ponto mais destoante deste estudo está no modelo **M3** quando utilizado como base somente os dados dos últimos três meses, essa acurácia baixa pode ser explicada pelo número baixo de observações utilizadas para a realização do treinamento do modelo que fez com que o modelo não alcançasse a devida distribuição de pesos as variáveis disponíveis.

É possível notar que ao não utilizar uma restrição de dados temos os modelos apresentando resultados mais estáveis apresentando acurácias para a base de teste sempre entre as melhores entre os conjuntos estudados.

Para este estudo foi realizado o treinamento com 80% dos eventos mais antigos de cada grupo estudado, enquanto os 20% de dados restantes (mais recentes) foi utilizado para teste.

5 CONCLUSÃO

Dado o objetivo de prever com alto nível de assertividade os vencedores de partidas profissionais do jogo *League of Legends (LoL)* para facilitar a tomada de decisão de apostadores deste tipo de mercado, podemos dizer que a solução proposta satisfaz fielmente este objetivo.

Durante a realização deste trabalho foi desenvolvida uma arquitetura capaz de prever os vencedores de jogos profissionais do título em análise com alto nível de acurácia e notifica os possíveis interessados em tempo real.

As acurácias dos modelos desenvolvidos foram satisfatórias e condizentes com os trabalhos relacionados ao tema, o que nos leva a crer que adicionando ao trabalho atual novas perspectivas é possível que se chegue a resultados ainda mais expressivos.

Algumas destas abordagens ficam como sugestão a trabalhos futuros para os interessados no tema.

5.1 PRÓXIMOS PASSOS/TRABALHOS FUTUROS

Após a realização deste trabalho fica claro outras abordagens que podem ser adicionadas ao mesmo de modo a aumentar acurácia/consistência dos modelos desenvolvidos.

Dentro dessas abordagens temos o próximo passo do estudo atual, que é a incorporação da análise das plataformas de apostas sobre o confronto que acontece, esta informação pode ser inferida através da *odd* que a casa de aposta concede para cada time previamente e durante a realização da partida.

Essas *odds* representam o quanto uma casa de aposta está disposta a recompensar o apostador por valor apostado. Esta conexão com as plataformas de *betting* não foi realizada neste trabalho, mas é um ponto-chave em um futuro desenvolvimento, possibilitando até mesmo automatizar possíveis apostas com base em graus de risco definidos pelo apostador.

Para melhorar a acurácia dos modelos, seria necessário a obtenção variáveis mais preditoras ainda não obtidas ou um aumento da quantidade de dados históricos. Ainda sobre este último item seria possível partir para duas abordagens em trabalhos futuros, são elas:

- Geração de dados artificialmente, simulando partidas profissionais com os dados que já existem.
- Utilização de jogos de jogadores casuais de *ranking* alto, estratégia utilizada em alguns estudos, referências no tema. Como por exemplo o trabalho *Win Prediction in Multi-Player Esports: Live Professional Match Prediction* (Hodge et al., 2019)

Já para a obtenção de novas variáveis fica a recomendação de adicionar novas fontes de dados a arquitetura da solução. Existem algumas APIs privadas no mercado que podem resultar em mais *features* e mais consistência do que a fonte disponibilizada pela produtora do jogo.

6 ANEXOS

6.1 LISTA DE VARIÁVEIS DOS MODELOS

- `game_external_ref`: Referência externa do jogo em questão
- `patch_version`: Versão do jogo no momento em que o mesmo é realizado
- `blue_team_ref`: Referência externa do time jogando no lado azul do mapa
- `red_team_ref`: Referência externa do time jogando no lado vermelho do mapa
- `blue_team_total_gold`: Total de recurso arrecadado pelo time azul no jogo
- `blue_team_inhibitors`: Total de inibidores destruídos pelo time azul no jogo
- `blue_team_towers`: Total de torres destruídas pelo time azul no jogo
- `blue_team_barons`: Total de *barons* destruídos pelo time azul no jogo
- `blue_team_total_kills`: Total de abates realizados pelo time azul no jogo
- `red_team_total_gold`: Total de recurso arrecadado pelo time vermelho no jogo
- `red_team_inhibitors`: Total de inibidores destruídos pelo time vermelho no jogo
- `red_team_towers`: Total de torres destruídas pelo time vermelho no jogo
- `red_team_barons`: Total de *barons* destruídos pelo time vermelho no jogo
- `red_team_total_kills`: Total de abates realizados pelo time vermelho no jogo
- `game_timestamp`: *Timestamp* em questão do jogo, importante para definir o momento do jogo ao que os dados se refere.
- `game_number`: Número do jogo, serve para diferenciar qual jogo de uma série o evento em questão se refere
- `team_a_side`: Definição de qual lado o time 'A' jogou, azul ou vermelho
- `team_b_side`: Definição de qual lado o time 'B' jogou, azul ou vermelho
- `team_a_record_wins`: Número de vitórias no campeonato em questão do time A
- `team_b_record_wins`: Número de vitórias no campeonato em questão do time B
- `team_a_record_losses`: Número de derrotas no campeonato em questão do time A

- `team_b_record_losses`: Número de derrotas no campeonato em questão do time B
- `best_of`: Se o jogo é de uma série MD1, MD2, MD3 ou MD5
- `red_team_dragons`: Número de dragões derrotados na partida pelo time do lado vermelho
- `blue_team_dragons`: Número de dragões derrotados na partida pelo time do lado azul
- `team_a_3_barons_mean_at25`: Média do número de *barons* aos 25 minutos de jogo do time A nos últimos 3 jogos
- `team_a_3_dragons_mean_at15`: Média do número de dragões aos 15 minutos de jogo do time A nos últimos 3 jogos
- `team_a_3_dragons_mean_at25`: Média do número de dragões aos 25 minutos de jogo do time A nos últimos 3 jogos
- `team_a_3_form_ratio`: Número de vitórias nos últimos 3 jogos do time A dividido por 3
- `team_a_3_gold_total_mean_at15`: Número de ouro médio aos 15 minutos de jogo dos últimos 3 jogos
- `team_a_3_gold_total_mean_at25`: Número de ouro médio aos 25 minutos de jogo dos últimos 3 jogos
- `team_a_3_inhibitors_mean_at15`: Número médio de inibidores destruídos nos últimos 3 jogos até os 15 minutos
- `team_a_3_inhibitors_mean_at25`: Número médio de inibidores destruídos nos últimos 3 jogos até os 25 minutos
- `team_a_3_kills_mean_at15`: Número médio de eliminações destruídos nos últimos 3 jogos até os 15 minutos
- `team_a_3_kills_mean_at25`: Número médio de eliminações destruídos nos últimos 3 jogos até os 25 minutos
- `team_a_3_towers_mean_at15`: Número médio de torres destruídas até os 15 minutos de jogos nos últimos 3 jogos
- `team_a_3_towers_mean_at25`: Número médio de torres destruídas até os 15 minutos de jogos nos últimos 3 jogos
- `team_a_5_barons_mean_at25`: Número médio de *barons* eliminados nos últimos 5 jogos até os 25 minutos

- `team_a_5_dragons_mean_at15`: Número médio de dragões eliminados nos últimos 5 jogos até os 15 minutos
- `team_a_5_dragons_mean_at25`: Número médio de dragões eliminados nos últimos 5 jogos até os 25 minutos
- `team_a_5_form_ratio`: Número de vitórias nos últimos 5 jogos do time A dividido por 5
- `team_a_5_gold_total_mean_at15`: Número de ouro médio aos 15 minutos de jogo dos últimos 5 jogos
- `team_a_5_gold_total_mean_at25`: Número de ouro médio aos 25 minutos de jogo dos últimos 5 jogos
- `team_a_5_inhibitors_mean_at15`: Número médio de inibidores destruídos nos últimos 5 jogos até os 15 minutos
- `team_a_5_inhibitors_mean_at25`: Número médio de inibidores destruídos nos últimos 5 jogos até os 25 minutos
- `team_a_5_kills_mean_at15`: Número médio de eliminações destruídos nos últimos 5 jogos até os 15 minutos
- `team_a_5_kills_mean_at25`: Número médio de eliminações destruídos nos últimos 5 jogos até os 25 minutos
- `team_a_5_towers_mean_at15`: Número médio de torres destruídas até os 15 minutos de jogos nos últimos 5 jogos
- `team_a_5_towers_mean_at25`: Número médio de torres destruídas até os 25 minutos de jogos nos últimos 5 jogos
- `team_b_3_barons_mean_at25`: Média do número de *barons* aos 25 minutos de jogo do time B nos últimos 3 jogos
- `team_b_3_dragons_mean_at15`: Média do número de dragões aos 15 minutos de jogo do time B nos últimos 3 jogos
- `team_b_3_dragons_mean_at25`: Média do número de dragões aos 25 minutos de jogo do time B nos últimos 3 jogos
- `team_b_3_form_ratio`: Número de vitórias nos últimos 3 jogos do time B dividido por 3
- `team_b_3_gold_total_mean_at15`: Número de ouro médio aos 15 minutos de jogo dos últimos 3 jogos

- team_b_3_gold_total_mean_at25: Número de ouro médio aos 25 minutos de jogo dos últimos 3 jogos
- team_b_3_inhibitors_mean_at15: Número médio de inibidores destruídos nos últimos 3 jogos até os 15 minutos
- team_b_3_inhibitors_mean_at25: Número médio de inibidores destruídos nos últimos 3 jogos até os 25 minutos
- team_b_3_kills_mean_at15: Número médio de eliminações destruídos nos últimos 3 jogos até os 15 minutos
- team_b_3_kills_mean_at25: Número médio de eliminações destruídos nos últimos 3 jogos até os 25 minutos
- team_b_3_towers_mean_at15: Número médio de torres destruídas até os 15 minutos de jogos nos últimos 3 jogos
- team_b_3_towers_mean_at25: Número médio de torres destruídas até os 15 minutos de jogos nos últimos 3 jogos
- team_b_5_barons_mean_at25: Número médio de *barons* eliminados nos últimos 5 jogos até os 25 minutos
- team_b_5_dragons_mean_at15: Número médio de dragões eliminados nos últimos 5 jogos até os 15 minutos
- team_b_5_dragons_mean_at25: Número médio de dragões eliminados nos últimos 5 jogos até os 25 minutos
- team_b_5_form_ratio: Número de vitórias nos últimos 5 jogos do time B dividido por 5
- team_b_5_gold_total_mean_at15: Número de ouro médio aos 15 minutos de jogo dos últimos 5 jogos
- team_b_5_gold_total_mean_at25: Número de ouro médio aos 25 minutos de jogo dos últimos 5 jogos
- team_b_5_inhibitors_mean_at15: Número médio de inibidores destruídos nos últimos 5 jogos até os 15 minutos
- team_b_5_inhibitors_mean_at25: Número médio de inibidores destruídos nos últimos 5 jogos até os 25 minutos
- team_b_5_kills_mean_at15: Número médio de eliminações destruídos nos últimos 5 jogos até os 15 minutos

- `team_b_5_kills_mean_at25`: Número médio de eliminações destruídos nos últimos 5 jogos até os 25 minutos
- `team_b_5_towers_mean_at15`: Número médio de torres destruídas até os 15 minutos de jogos nos últimos 5 jogos
- `team_b_5_towers_mean_at25`: Número médio de torres destruídas até os 25 minutos de jogos nos últimos 5 jogos
- `participant_id_player_{player_number}`: Número do jogador da partida que identifica a posição do mesmo em cada equipe, de 1 a 10
- `champion_name_player_{player_number}`: Nome do campeão escolhido pelo jogador
- `role_player_{player_number}`: Posição do jogador do jogo
- `level_player_{player_number}`: Nível do jogador no jogo corrente
- `kills_player_{player_number}`: Número de eliminações do jogador no jogo
- `deaths_player_{player_number}`: Número de mortes do jogador no jogo
- `assists_player_{player_number}`: Número de assistências do jogador no jogo
- `total_gold_earned_player_{player_number}`: Total de ouro recebido pelo jogador no jogo
- `creep_score_player_{player_number}`: Número de *minions* derrotados pelo jogador no jogo
- `kill_participation_player_{player_number}`: Percentual de participação nas eliminações do time do jogador em questão
- `champion_damage_share_player_{player_number}`: Quantidade de dano efetuado a campeões inimigos pelo jogador
- `wards_placed_player_{player_number}`: Número de sentinelas posicionadas pelo jogador no jogo
- `wards_destroyed_player_{player_number}`: Número de sentinelas destruídas pelo jogador no jogo
- `game_winner`: Vencedor do jogo

REFERÊNCIAS

- Almeida, C. E. M., Correia, R. C. M., Eler, D. M., Olivete-Jr, C., Garci, R. E., Scabora, L. C. e Spadon, G. (2017). Prediction of winners in moba games. Em *2017 12th Iberian Conference on Information Systems and Technologies (CISTI)*, páginas 1–6.
- CBeS (2019). O que são os *eSports*? <http://cbesports.com.br/esports/esports-o-que-sao/#o-que-e-esports>. Acessado em 09/08/2021.
- Esporte, G. (2021). Lol: skins, runas, personagens; o que é e tudo sobre o moba da riot. <https://ge.globo.com/esports/lol/noticia/lol-skins-runas-personagens-o-que-e-e-tudo-sobre-o-moba-da-riot-games.ghtml>. Acessado em 17/07/2021.
- Games, R. (2021). Como jogar *League of Legends*. <https://br.leagueoflegends.com/pt-br/how-to-play/>. Acessado em 17/07/2021.
- Hodge, V., Devlin, S., Sephton, N., Block, F., Drachen, A. e Cowling, P. (2017). Win prediction in esports: Mixed-rank match prediction in multi-player online battle arena games.
- Hodge, V. J., Devlin, S. M., Sephton, N. J., Block, F. O., Cowling, P. I. e Drachen, A. (2019). Win prediction in multi-player esports: Live professional match prediction. *IEEE Transactions on Games*.
- Kim, D.-H., Lee, C. e Chung, K.-S. (2020). A confidence-calibrated moba game winner predictor. páginas 622–625.
- LeagueOfGraphs (2021). Leagueofgraphs. <https://www.leagueofgraphs.com/pt/rankings/game-durations>]. Acessado em 05/12/2021.
- Makarov, I., Savostyanov, D., Litvyakov, B. e Ignatov, D. I. (2018). Predicting winning team and probabilistic ratings in “dota 2” and “counter-strike: Global offensive” video games. Em van der Aalst, W. M., Ignatov, D. I., Khachay, M., Kuznetsov, S. O., Lempitsky, V., Lomazova, I. A., Loukachevitch, N., Napoli, A., Panchenko, A., Pardalos, P. M., Savchenko, A. V. e Wasserman, S., editores, *Analysis of Images, Social Networks and Texts*, páginas 183–196, Cham. Springer International Publishing.
- OddsShark (2018). O que são odds. <https://www.odsshark.com/br/como-apostar/o-que-sao-odds>. Acessado em 09/08/2021.
- Povoleri, B. (2021). Moba: o que é, características e games de sucesso. <https://www.theenemy.com.br/esports/moba-o-que-e-caracteristicas-e-principais-jogos-lol-dota>. Acessado em 09/08/2021.

- Rejane, S. (2021). Apostas esportivas no brasil: Entenda como funciona. <https://jus.com.br/artigos/92170/apostas-esportivas-no-brasil-entenda-como-funciona>. Acessado em 09/08/2021.
- Saidat Giwa-Osagie, S. B. (2021). Esports & games streaming: Emerging opportunities & market forecasts 2021-2025. <https://www.juniperresearch.com/researchstore/content-digital-media/esports-games-streaming-research-report>.
- Semenov, A., Romov, P., Korolev, S., Yashkov, D. e Neklyudov, K. (2017). Performance of machine learning algorithms in predicting game outcome from drafts in dota 2. Em Ignatov, D. I., Khachay, M. Y., Labunets, V. G., Loukachevitch, N., Nikolenko, S. I., Panchenko, A., Savchenko, A. V. e Vorontsov, K., editores, *Analysis of Images, Social Networks and Texts*, páginas 26–37, Cham. Springer International Publishing.
- Tartaglia, R. (2019). League of legends: do arauto ao barão, saiba importância e duração de cada buff da selva. <https://sportv.globo.com/site/esportv/lol/noticia/league-of-legends-do-arauto-ao-barao-saiba-importancia-e-duracao-de-cada-buff-da-selva.ghtml>. Acessado em 17/07/2021.
- Wang, N., Li, L., Xiao, L., Yang, G. e Zhou, Y. (2018). Outcome prediction of dota2 using machine learning methods. Em *Proceedings of 2018 International Conference on Mathematics and Artificial Intelligence*, ICMAI '18, página 61–67, New York, NY, USA. Association for Computing Machinery.
- Wang, T. (2018). Predictive analysis on esports games: A case study on league of legends (lol) esports tournaments. Dissertação de Mestrado, University of North Carolina at Chapel Hill.
- Wang, W. (2016). Predicting multiplayer online battle arena (moba) game outcome based on hero draft data. Dissertação de Mestrado, Dublin, National College of Ireland.
- Wijman, T. (2021). Global games market to generate 175.8billionin2021; despiteaslightdecline, themarketisontracktosurpass200 billion in 2023. <https://newzoo.com/insights/articles/global-games-market-to-generate-175-8-billion-in-2021-despite-a-slight-decline-the-market-is-on-track-to-surpass-200-billion-in-2023/>.
- Wikipedia (2021a). Arena de batalha multijogador em linha. https://pt.wikipedia.org/wiki/Arena_de_batalha_multijogador_em_linha. Acessado em 21/11/2021.
- Wikipedia (2021b). Esporte eletrônico. https://pt.wikipedia.org/wiki/Esporte_eletr%C3%B4nico. Acessado em 09/08/2021.

- Yang, P., Harrison, B. E. e Roberts, D. L. (2014). Identifying patterns in combat that are predictive of success in MOBA games. Em Mateas, M., Barnes, T. e Bogost, I., editores, *Proceedings of the 9th International Conference on the Foundations of Digital Games, FDG 2014, Liberty of the Seas, Caribbean, April 3-7, 2014*. Society for the Advancement of the Science of Digital Games.
- Yang, Y., Qin, T. e Lei, Y.-H. (2016). Real-time esports match result prediction.
- Yang, Z., Wang, Y., Li, P., Lin, S., Shi, S. e Huang, S.-L. (2020). Predicting events in moba games: Dataset, attribution, and evaluation.