**UNIVERSIDADE DE SOROCABA**

**PRÓ-REITORIA ACADÊMICA**

**CURSO DE CIÊNCIA DA COMPUTAÇÃO**

**Danilo de Lucas Moraes Dias**

**APRENDIZADO DE MÁQUINA APLICADO Á RESOLUÇÃO DE PROBLEMAS DE DECISÃO**

**Sorocaba**

**2016**

**Danilo de Lucas Moraes Dias**

**APRENDIZADO DE MÁQUINA APLICADO Á RESOLUÇÃO DE PROBLEMAS DE DECISÃO**

Trabalho de conclusão do curso de graduação apresentado na Universidade de Sorocaba como requisito parcial para a obtenção do título de Bacharel em Ciência da Computação.

Orientador: Fernando Cesar Miranda

**Sorocaba/SP**

**2016**

**RESUMO**

Os jogos eletrônicos sempre estiveram evoluindo no quesito gráfico, apresentando cada vez mais detalhes em texturas, luzes e sombras mais realistas, além de objetos com detalhes geométricos fotorrealistas. Esta evolução também se estendeu à complexidade dos sistemas que compõem um jogo, trazendo experiências mais imersivas aos jogadores como, por exemplo, inimigos que aprendem o padrão do jogador e se tornam mais difíceis de serem derrotados. Por conta disso, a evolução nos jogos propiciou que fossem desenvolvidas técnicas para aperfeiçoar a implementação de algoritmos inteligentes neste contexto. Trabalhos na literatura demonstraram avanço no desenvolvimento de algoritmos capazes de adaptar suas ações através da utilização de métodos como programação genética e redes neurais, os quais demandam uma quantidade significativa de tempo para serem processados e efetivamente gerar uma resposta em tempo hábil ao sistema. Para contornar essa dificuldade, este trabalho adota a estratégia de combinar métodos de busca e classificação através de algoritmos capazes de desenvolver conhecimento a respeito do contexto de um problema e gerar soluções em tempo de execução. Os resultados apresentados foram validados estatisticamente e indicaram que o método proposto obteve performance superior aos métodos que utilizam do treinamento de redes neurais para a resolução de problemas.

**Palavras-chaves:** Inteligência Artificial; Busca; Classificação; Aprendizado de Máquina; Jogos eletrônicos.

**ABSTRACT**

The Games have always been evolving…

**Key-words:** artificial intelligence; search; classification; machine learning; games.

**LISTA DE ILUSTRAÇÕES**

**LISTA DE ABREVIATURAS E SIGLAS**

IA Inteligência Artificial

NPC Non-player Character

NEAT Evolving Neural Networks through Augmenting Topologies

SMW Super Mario World

RNA Redes Neurais Artificiais

SNES Super Nintendo Entertainment System

TAS Tool Assisted Speedrun

SUMÁRIO

[**1** **INTRODUÇÃO** 5](#_Toc461806109)

[**2** **INTELIGÊNCIA ARTIFICIAL EM JOGOS** 7](#_Toc461806110)

[**3** **Métodos de Classificação** 8](#_Toc461806111)

[**3.1** **Métodos Baseados em Regra** 9](#_Toc461806112)

[**3.2** **Métodos Probabilísticos** 9](#_Toc461806113)

[**3.3** **Métodos SVM** 9](#_Toc461806114)

[**3.4** **Métodos Baseados em Instancia** 9](#_Toc461806115)

[**3.5** **Redes Neurais** 9](#_Toc461806116)

[**3.6** **Árvores de Decisão** 9](#_Toc461806117)

[**3.1.1** **Função de Entropia** 10](#_Toc461806118)

[**3.1.2** **Função de Ganho** 10](#_Toc461806119)

[**REFERÊNCIAS** 11](#_Toc461806120)

1. **INTRODUÇÃO**

A medida que o processamento, os gráficos e o realismo dos jogos aumentam, a exigência dos jogadores por uma experiência mais imersiva também cresce. Estes fatores, criaram a necessidade de que os elementos controlados pelo computador, tais como obstáculos e NPC’s (*non-player characters,* personagens não controlados pelo jogador), tenham não somente reações a interações do jogador, mas também aprendam com o decorrer do jogo para adaptar suas ações. Alguns jogos possuem mecanismos que comportam a utilização de métodos de aprendizado para a implementação de inteligências artificiais (IA), o que indica que elas não precisam ser programadas manualmente (CHAMPANDARD, 2004). Partindo deste princípio, os desenvolvedores buscam criar técnicas e métodos para a implementação de algoritmos que se moldem utilizando os dados gerados no decurso do jogo.

Trabalhos na literatura indicam ser promissora a aplicação de métodos de aprendizado de máquina para a implementação de inteligências artificiais em jogos (STANLEY, 2002; MIIKKULAINEN, 2002). Estes métodos geram dados que são utilizados para o treinamento de redes neurais[[1]](#footnote-1), que posteriormente serão utilizadas para a resolução dos problemas. Contudo, algumas destas implementações demonstram-se lentas, pois utilizam técnicas como programação genética[[2]](#footnote-2) para otimização de soluções, sendo necessárias várias execuções para que uma solução que atenda ao problema seja alcançada.

Em um jogo, um problema pode ser considerado como uma situação que é apresentada ao jogador ou ao computador, onde as variáveis são os atributos dos inimigos e obstáculos, e o contexto é a relação entre as variáveis e o elemento controlado pelo jogador ou computador. Quando se tem um problema que não é conhecido, inferir dados sobre o contexto pode ser uma boa forma de chegar a uma solução para o problema. Para obter estes dados podem ser utilizados mecanismos de busca, que farão a varredura do contexto do problema a procura de soluções, gerando dados sobre o mesmo. Contudo, independente da implementação, seria necessário gerar dados específicos para cada problema proposto, o que não é o propósito de uma IA de acordo com Tanimoto (TANIMOTO, 1987). Portanto é necessária uma forma de generalizar soluções para que as mesmas possam ser aplicadas a outros problemas parecidos, sem que necessariamente seja preciso refazer a busca e gerar novos dados. Uma forma de conseguir esta generalização é por meio da aplicação de técnicas de aprendizagem supervisionada, que irão utilizar os dados gerados pela busca para o seu treinamento, e com isso generalizar soluções para os problemas propostos.

Para a busca por soluções e geração de dados destaca-se a utilização de algoritmos recursivos que utilizam processos heurísticos para estimar a procedência positiva ou negativa de uma iteração (GHADERI, 2009). Trata-se de um método de refinamento de busca por força bruta, que faz a varredura do contexto do problema a procura de soluções, utilizando heurísticas para ignorar soluções errôneas e com isso gerar dados específicos sobre os problemas solucionados. A generalização de soluções demonstra-se eficiente com a utilização de métodos de classificação que consomem dados para o treinamento de árvores de decisão (FUSSELL, 2012). A árvore gerada é ajustada mediante aos valores fornecidos em sua instancia, classificando da melhor forma possível uma dada observação, com base nos valores disponíveis na base durante a geração da árvore.

Nesse cenário, onde é preciso que para um dado problema seja encontrada uma solução, e posteriormente possa ser generalizada para outros problemas, a utilização dos métodos de busca e classificação, em conjunto, mostra-se viável. Com a aplicação do método de busca, é possível gerar uma base de dados com soluções especificas para problemas resolvidos. A partir desta base, o método de classificação possibilita a generalização das soluções para problema não tratados, gerando a melhor classificação possível para problemas semelhantes e possíveis boas soluções para problemas muito diferentes.

**Objetivos e Contribuições**

O objetivo deste trabalho é apresentar técnicas para a implementação de inteligência artificial em jogos por meio da geração e manipulação de bases de conhecimento, através da utilização de conceitos de aprendizado supervisionado e otimização de busca.

Dentre as contribuições oferecidas neste trabalho, destacam-se:

1. Demonstração da utilização de uma base de dados gerada pelo método de backtracking para o treinamento de árvores de decisão;
2. Criação de uma biblioteca para o auxílio na implementação de algoritmos de aprendizado de máquina;
3. Geração de bases de conhecimento que podem ser utilizadas para treinamento de outros algoritmos de classificação.

**Organização**

Este manuscrito apresenta a seguinte estrutura:

* No Capítulo 2, é introduzida a utilização de inteligência artificial em jogos e os principais trabalhos encontrados na literatura.
* No Capítulo 3, são introduzidos os conceito de aprendizado de máquina e descritos os métodos de classificação envolvidos no trabalho.
* No Capítulo 4, são abordadas as técnicas de otimização de busca para a geração de bases de conhecimento.
* No Capítulo 5, é apresentado o algoritmo resultante das pesquisas realizadas neste trabalho, assim como a biblioteca desenvolvida para facilitar a implementação do mesmo.
* No Capítulo 6, são apresentados os experimentos realizados e os resultados obtidos pela pesquisa.
* Finalmente, no Capítulo 7, são dadas as conclusões e orientações para trabalhos futuros.

1. **INTELIGÊNCIA ARTIFICIAL EM JOGOS**
2. **Métodos de Classificação**

Os métodos de classificação exercem um importante papel no mercado de tomada de decisões, classificando as informações disponíveis utilizando algum critério para tal (KIANG, 2002). A implementação de inteligências artificiais que precisam desenvolver-se durante a execução do programa, normalmente é feita através de métodos de aprendizado de máquina. Geralmente o campo do aprendizado de máquina é dividido em três grandes grupos: aprendizagem supervisionada, aprendizagem não-supervisionada e aprendizagem por reforço (RUSSEL; NORVIN, 2003).

* Aprendizagem supervisionada: o programa recebe amostras de entradas e suas respectivas saídas, e gera uma função que pode ser utilizada para classificar observações que não se conhece o resultado.
* Aprendizagem não-supervisionada: o programa trata de identificar os padrões do problema com base nas entradas, sem que nenhuma saída especifica seja disponibilizada.
* Aprendizagem por reforço: o programa interage com o ambiente do problema e aprende conforme o explora.

O grupo a ser utilizado depende do tipo dos dados disponibilizados pelo problema. O aprendizado supervisionado se mostra o mais adequado ao contexto do trabalho em questão, pois os dados gerados por ele são tipicamente de problemas de classificação **(explicar)**. O contexto do problema tratado no trabalho apresenta uma base de observações solucionadas que pode ser utilizada para o treinamento do programa.

Na literatura, são utilizados métodos de classificação para as mais diversas tarefas. Em 2009, um artigo disponibilizado pelo *Jornal of Convergence Information Technology* tratou de fazer uma comparação dos métodos de classificação baseando-se no tipo dos atributos e no tamanho das amostras (ENTEZARI-MALEKI, 2009). Os resultados do trabalho demostram os melhores métodos a serem utilizados mediante as circunstancias do problema.

O métodos mais comuns utilizados para classificação de dados são as arvores de decisão, métodos baseados em regra, métodos probabilísticos, métodos SVM, métodos baseados em instancia e redes neurais (AGGARWAL, 2015). Na seção a seguir será apresentada uma introdução sucinta a respeito dos métodos citados, com um enfoque partícula nos métodos baseados em árvore, por estar dentro dos métodos utilizados neste trabalho.

* 1. **Árvores de Decisão**

Indução por árvores de decisão é um dos mais simples e assertivos métodos de aprendizado de máquina (RUSSEL; NORVIG, 2003). Se trata de um algoritmo de aprendizado de fácil implementação e que apresenta um alto nível de precisão. No decorrer do capítulo serão descritas as técnicas utilizadas para geração e manipulação das árvores.

Na geração de uma árvore de decisão, são introduzidas situações que são descritas por meio de um conjunto de propriedades ou atributos e seus respectivos resultados, que são utilizados para montar os nós de condição e os ramos e folhas de resultado. Cada nó da árvore corresponde a uma condição composta por uma propriedade e um objeto de teste. A árvore apresenta um resultado para cada condição, seja ela verdadeira ou falsa, podendo ser um resultado final, representado por uma folha, ou um ramo, que levará a outros nós a serem testadas. Após sua geração, recebe uma observação que apresenta somente um conjunto de propriedades, que será utilizado como objeto de teste para percorrer os nós da árvore e resultar em uma saída booleana (sim ou não). Implementações com saídas mais abrangentes também podem ser representadas, se for necessário que o resultado esteja presente em um conjunto.

É importante que durante a geração da árvore sejam escolhidos os melhores atributos para montar os ramos, pois uma escolha incorreta dos atributos pode acarretar a construção de uma árvore ineficiente. O objetivo então é maximizar a homogeneidade/pureza de cada conjunto por meio de heurísticas, sendo que o melhor atributo do conjunto é o que possui o maior nível de pureza dentre os demais. Para tal, o conjunto de atributos é iterado e são utilizados métodos heurísticos para determinar a homogeneidade do conjunto de observações atual da iteração. A cada nova iteração, o processo é repetido excluindo do conjunto os atributos que já foram utilizados anteriormente. Os atributos que apresentam a maior pureza são priorizados, pois quando a condição de um nó é gerada com base neles a classificação de observações é otimizada, pois os ramos serão divididos de forma a terem muitas soluções de um lado e poucas do outro, fazendo com que a maior parte das soluções inconsistentes sejam desconsideradas. Com a pureza dos atributos sendo levada em consideração, a árvore tenderá a ser dividida de forma a priorizar os atributos que menos mudaram durante o treinamento, assim diminuindo o tamanho da árvore e ignorando grande parte dos resultados incoerentes durante a classificação de uma dada observação.

Para a construção das arvores de decisão são utilizadas as técnicas a seguir.

* + 1. **Função de Entropia**

A função de entropia…

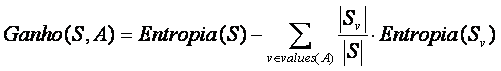
http://www.isw.rwth-aachen.de/bock/Publikationen/A38-Bock-ProbabModelsinCluster-CSDA-1996.pdf

http://web.tecnico.ulisboa.pt/ana.freitas/bioinformatics.ath.cx/bioinformatics.ath.cx/uploads/RTEmagicC_arv_dec7.gif.gif

http://web.tecnico.ulisboa.pt/ana.freitas/bioinformatics.ath.cx/bioinformatics.ath.cx/uploads/RTEmagicC_arv_dec8.gif.gif



* + 1. **Função de Ganho**



* 1. **Métodos Baseados em Regra**
  2. **Métodos Probabilísticos**
  3. **Métodos SVM**
  4. **Métodos Baseados em Instancia**
  5. **Redes Neurais**

**REFERÊNCIAS**

STANLEY, Kenneth O.; MIIKKULAINEN, Risto. **Evolving Neural Networks through Augmenting Topologies.** Massachusetts Institute of Technology, 2002. Disponível em: <<http://goo.gl/TccvC>>

KISHIMOTO, André. **Inteligência Artificial em Jogos Eletrônicos.** 11 p. Disponível em: <<http://goo.gl/CS45lZ>>

CHAMPANDARD, Alex J. **AI Game Development: Synthetic Creatures with Learning and Reactive Behaviors.** 8 p. Disponível em: <<http://goo.gl/Cj8khW>>

TANIMOTO, Steven L. **The Elements of Artificial Intelligence.** Washington: Computer Science Press, 1987. Disponível em: <<http://goo.gl/O9n6f6>>

GHADERI, Hojjat. **CSC384: Introduction to Artificial Intelligence.** University of Toronto, 2009. 5 cap – Backtracking Search. Disponível em: <<http://goo.gl/8UhmeO>>

FUSSELL, Don. **AI – Decision Trees and Rules Systems.** University of Texas at Austin, 2012. Disponível em: <<http://goo.gl/jyp8yK>>

RUSSEL, Stuart J.; NORVIG, Peter. **Artificial Intelligence: A Modern Approach. 2. ed.** Pearson Education, 2003.

AGGARWAL, Charu C. **Data Classification: Algorithms and Applications**.IBM T. J. Watson Research Center, Yorktown Heights, New York, USA. Disponível em: <<https://goo.gl/4EGygO>>

KIANG, Melody Y. **A Comparative Assessment of Classification Methods.** Information System Department, College of Business Administration, California State University, 2002. Disponível em: <<https://goo.gl/LcBgHI>>

ENTEZARI-MALEKI, R.; RAZAEI, A.; MINAEI-BIDGOLI, B. **Comparison of Classification Methods Based on the Type of Attributes and Sample Size.** Department of Computer Engeneering, Iran University of Science & Technology, Tehran, Iran, 2009. Disponível em: <<https://goo.gl/wc2ltw>>

**ANEXOS**

1. Redes neurais artificiais são modelos computacionais que abstraem o funcionamento do sistema nervoso central de um animal. [↑](#footnote-ref-1)
2. Programação genética é uma técnica de programação autônoma que aplica princípios da evolução biológica para manipular soluções. [↑](#footnote-ref-2)