



UNIVERSIDADE FEDERAL DO PARÁ
INSTITUTO DE CIÊNCIAS EXATAS E NATURAIS
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

Gilberto Nerino de Souza Junior

**Análise comportamental utilizando aprendizado de máquina para auxiliar
o ensino da leitura e escrita aplicado em um jogo digital**

Belém
2013

Gilberto Nerino de Souza Junior

**Análise comportamental utilizando aprendizado de máquina para auxiliar
o ensino da leitura e escrita aplicado em um jogo digital**

Dissertação de Mestrado apresentada para
obtenção do grau de Mestre em Ciência
da Computação.

Programa de Pós-Graduação em Ciência
da Computação.

Instituto de Ciências Exatas e Naturais.
Universidade Federal do Pará.

Área de concentração em Sistemas de
Computação, linha de pesquisa Sistemas
Inteligentes.

Orientador Prof. Dr. Dionne Cavalcante
Monteiro.

Belém

2013

Gilberto Nerino de Souza Junior

**Análise comportamental utilizando aprendizado de máquina para auxiliar
o ensino da leitura e escrita aplicado em um jogo digital**

Dissertação de Mestrado apresentada para
obtenção do grau de Mestre em Ciência
da Computação. Programa de Pós-
Graduação em Ciência da Computação.
Instituto de Ciências Exatas e Naturais.
Universidade Federal do Pará

Data da aprovação: Belém-PA. 22 - 02 - 2013

Banca Examinadora

Prof. Dr. Dionne Cavalcante Monteiro
Instituto de Computação – UFPA - Orientador

Prof. Dr. Claudomiro de Souza de Sales Junior
Instituto de Computação – UFPA – Membro

Prof. Dr. Ádamo Lima de Santana
Instituto de Engenharia Elétrica e Computação – UFPA – Membro

Belém
2013

Dedico aos meus pais, meus irmãos, minha namorada, meu orientador,
minha eterna gratidão.

Agradecimento

À Universidade Federal do Pará.

Ao Professor e Amigo Dionne Monteiro por ter me guiado de forma correta e ter me corrigido nas horas que eu estava errado.

Aos amigos do Laboratório: Adalberto Bosco, Hyago Pinheiro, Pedro Afonso e Ellton Sales pelo incentivo e apoio técnico.

Ao amigo da Universidade Federal de São Carlos, Leonardo Marques, pelo apoio na área da psicologia experimental.

A minha família, em que me incentivou e ajudou de todas as formas nessa jornada.

A todos aqueles que direta ou indiretamente contribuíram para a realização deste trabalho.

A Deus, por ter guiado meus caminhos.

“No crepúsculo da vida, seremos julgados pelo amor”.

S. J. C

RESUMO

Diversas pesquisas possuem foco em características do aprendizado de máquina e sua aplicação em campos como a psicologia, educação e criação de jogos digitais. O programa de ensino de leitura e escrita “Aprendendo a Ler e Escrever em Pequenos Passos” (ALEPP) concebido por psicólogos da UFSCar é uma metodologia de ensino que tem por objetivo desenvolver o ensino individualizado. Alguns trabalhos geraram produtos informatizados dessa metodologia, porém, não utilizaram uma análise computacional do aprendizado do aluno a fim de prever e classificar características do aprendizado. Este trabalho é um estudo aplicado na área da psicologia experimental baseado no ALEPP com o objetivo de agregar uma máquina de aprendizado para prever o acerto ou erro de tarefas de ensino e classificar o aprendizado das palavras ensinadas a fim de analisar o comportamento do aluno, com ou sem histórico de fracasso. Para isso foi necessário identificar os fatores de dificuldade das tarefas de ensino e o tipo de problema de aprendizado de máquina a ser implementado. A técnica utilizada foi a regressão logística a fim de solucionar o problema de aprendizado supervisionado aliado com outros parâmetros estatísticos vindouros da base de dados. A máquina de aprendizado desenvolvida foi agregada a um jogo digital para tornar o ensino lúdico. As simulações dos comportamentos dos alunos obtiveram resultados satisfatórios de acordo com a avaliação dos especialistas. O jogo pretende auxiliar professores, pedagogos e psicólogos que trabalham com alunos no processo de aprendizado em fase de alfabetização além de abrir portas para futuros trabalhos como a geração de tarefas personalizadas.

PALAVRAS-CHAVE: Análise Comportamental, Aprendizado de Máquina, Jogos Educacionais, Leitura e Escrita.

ABSTRACT

Various studies have focused on features of machine learning and its application in fields as psychology, education and creation of digital games. The program of instruction in reading and writing "Learning to Read and Write in Small Steps" or "Aprendendo a Ler e Escrever em Pequenos Passos" in portuguese (ALEPP) designed by psychologists UFSCar is a teaching methodology that aims to develop individualized teaching. Some work computer generated products of this methodology, however, did not use a computer analysis of student learning to predict and classify learning characteristics. This work is an applied study in the field of experimental psychology based on ALEPP with the goal of adding a machine learning to predict the rightness or wrongness of teaching tasks and classify the learning of the words taught to analyze the behavior of the student, with or without a history of failure. This required identifying the factors of difficulty of the tasks of teaching and type of machine learning problem to be implemented. The technique was used logistic regression to solve the problem of supervised learning together with other statistical parameters coming from the database. The machine learning was developed to aggregate a digital game to make learning entertaining. The simulations of the behavior of students achieved satisfactory results according to the expert reviews. The game aims to help teachers, pedagogues and psychologists who work with students in the learning phase of literacy and open doors for future work as the generation of custom tasks.

KEYWORDS: Behavioral Analysis, Machine Learning, Educational Games, Reading and Writing.

LISTA DE FIGURAS

FIGURA 1 – ESBOÇO DA METODOLOGIA UTILIZADA.....	22
FIGURA 2 - PROGRAMA GERADOR DE DIFICULDADE DAS TENTATIVAS INTEGRADO COM A REGRESSÃO LOGÍSTICA.	23
FIGURA 3 – ESQUEMA DE REPRESENTAÇÃO DA CLASSE ESTÍMULOS E RESPOSTAS EM CONFORMIDADE COM SUAS RELAÇÕES	27
FIGURA 4 – FLUXO DA DISCRIMINAÇÃO CONDICIONAL NO PROGRAMA DE ENSINO	29
FIGURA 5 – REPRESENTAÇÃO EM ALTO NÍVEL DE UMA MÁQUINA DE APRENDIZADO	34
FIGURA 6 – VISÃO HIERÁRQUICA DO APRENDIZADO INDUTIVO	35
FIGURA 7 – ESTRUTURA DE UMA SESSÃO DE ENSINO.....	39
FIGURA 8 – EXEMPLO DE UMA TAREFA DE ENSINO DO GEIC ONDE A CRIANÇA TEM QUE MONTAR A PALAVRA “TATU”	39
FIGURA 9 – UMA CENA DO JOGO ALE-RPG DESENVOLVIDO PARA AUXILIAR NO APRENDIZADO DA LEITURA E ESCRITA DE CRIANÇAS.	40
FIGURA 10 – CENA DE UMA TAREFA DE ENSINO DO JOGO ALE-RPG. AO LADO UM QUADRO DE ATRIBUTOS DO JOGADOR.	40
FIGURA 11 - JOGO GRAPHO GAME DESENVOLVIDO PARA AUXILIAR O APRENDIZADO DE FONEMAS E ESCRITA PARA CRIANÇAS COM DISLEXIA E DIFICULDADE NO ENSINO.	41
FIGURA 12 – UM MINI-JOGO ONDE HÁ A EMISSÃO DE FONEMA E A CRIANÇA DEVE CLICAR NA SÍLABA CORRESPONDENTE.	41
FIGURA 13 - JOGO <i>DARWIN KOMBAT</i> É UM SIMULADOR DE JOGOS DE ESTRATÉGIA DESENVOLVIDO PARA EXPERIMENTOS COM MÁQUINAS DE APRENDIZADO. A) PRIMEIRA ZONA DO JOGO.	42
FIGURA 14 - O JOGO COMERCIAL <i>BLACK AND WHITE</i> É CONSIDERADO UM JOGO IMPORTANTE NA ÁREA DE VIDA ARTIFICIAL.....	43
FIGURA 15 – PEGAR-E-IR NO SIMULADOR DE JOGOS DE FUTEBOL <i>TAO OF SOCCER</i> COMO UM COMPORTAMENTO CLASSIFICADO.	45
FIGURA 16 – ALGUNS MINIJOGOS UTILIZADOS NESTA PESQUISA.	46
FIGURA 17 – FOTO DO JOGO <i>OPEN SOURCE SUPER TUXKART</i> UTILIZADO NESTA PESQUISA.....	46
FIGURA 18–ESQUEMA COM OS DIVERSOS DADOS QUE COMPÕEM UMA TAREFA DE ENSINO.	51
FIGURA 19 – EXEMPLO DE UMA TAREFA REFERENTE À LEITURA.	53
FIGURA 20 – EXEMPLO DE UMA TAREFA REFERENTE À ESCRITA.	55
FIGURA 21 – FLUXOGRAMA GERAL UTILIZADO NO SISTEMA COMO UM TODO.	58
FIGURA 22 – REPRESENTAÇÃO DO PROCESSO DE APRENDIZADO DE MÁQUINA NO PROGRAMA E ENSINO.....	60
FIGURA 23 - MÁQUINA DE APRENDIZADO JUNTAMENTE COM A LÓGICA DE PRIMEIRA ORDEM.....	66
FIGURA 24 - ESTRUTURA DO PROTÓTIPO DA MÁQUINA DE APRENDIZADO.	67
FIGURA 25 – INTERAÇÃO COM PROTOTIPAGEM RÁPIDA PARA JOGOS.....	71
FIGURA 26 – AMARU E ÚRAMA.	71
FIGURA 27 – <i>BACKGROUNDS</i> DOS CENÁRIOS DO JOGO.....	72
FIGURA 28 – <i>GAMEPLAY</i> DO JOGO COM TAREFAS REFERENTES A LEITURA.	72
FIGURA 29 – <i>GAMEPLAY</i> DO JOGO COM TAREFAS REFERENTES A ESCRITA.....	73
FIGURA 30 – VISÃO MACRO DO PROJETO.....	75

FIGURA 31 – INTERFACE DO PROGRAMA RELATÓRIO AMARU.....	76
FIGURA 32 – RELATÓRIO GERADO PELO PROGRAMA.	77

LISTA DE GRÁFICOS

GRÁFICO 1 - ASPECTO DE UMA CURVA SIGMÓIDE OU LOGÍSTICA.....	62
GRÁFICO 2 - GRÁFICO DOS DADOS DE TREINO DA PRIMEIRA SIMULAÇÃO.....	80
GRÁFICO 3 - GRÁFICO DOS DADOS DE TREINO DA SEGUNDA SIMULAÇÃO.....	81
GRÁFICO 4 – CONVERGÊNCIA DO PARÂMETRO DE APRENDIZADO θ DAS TRÊS SIMULAÇÕES PARA A PALAVRA BOLO.	84
GRÁFICO 5 – CONVERGÊNCIA DO PARÂMETRO DE APRENDIZADO θ DAS TRÊS SIMULAÇÕES PARA A PALAVRA TATU.	85
GRÁFICO 6 – CONVERGÊNCIA DO PARÂMETRO DE APRENDIZADO θ DAS TRÊS SIMULAÇÕES PARA A PALAVRA APITO.	86
GRÁFICO 7 – CONVERGÊNCIA DO PARÂMETRO DE APRENDIZADO θ DAS TRÊS SIMULAÇÕES PARA A PALAVRA TOMATE.....	86
GRÁFICO 8 – CONVERGÊNCIA DO PARÂMETRO DE APRENDIZADO θ DAS TRÊS SIMULAÇÕES PARA A PALAVRA MULETA	87
GRÁFICO 9 – PREVISÃO DA PROBABILIDADE DE ACERTO EM UM INTERVALO DE DIFICULDADE PARA A PALAVRA BOLO.	88
GRÁFICO 10 – PREVISÃO DA PROBABILIDADE DE ACERTO EM UM INTERVALO DE DIFICULDADE PARA A PALAVRA TATU.	89
GRÁFICO 11 – PREVISÃO DA PROBABILIDADE DE ACERTO EM UM INTERVALO DE DIFICULDADE PARA A PALAVRA APITO.	89
GRÁFICO 12 – PREVISÃO DA PROBABILIDADE DE ACERTO EM UM INTERVALO DE DIFICULDADE PARA A PALAVRA TOMATE.....	90
GRÁFICO 13 – PREVISÃO DA PROBABILIDADE DE ACERTO EM UM INTERVALO DE DIFICULDADE PARA A PALAVRA MULETA.	91
GRÁFICO 14 - TAREFAS PARA O ENSINO DA PALAVRA “BOLO” NO COMPORTAMENTO DÉFICIT DE APRENDIZADO.	92

LISTA DE TABELAS

TABELA 1 – PARTICIPANTES DO PROJETO.	23
TABELA 2 – REPRESENTAÇÃO DE TIPO DE ESTÍMULOS IDENTIFICADO COMO LETRAS.	27
TABELA 3 – EXEMPLOS DE DIFERENTES RELAÇÕES ENTRE TAREFAS DO TIPO MTS.	30
TABELA 4 – TRÊS EXEMPLOS DE DIFERENTES RELAÇÕES ENTRE TAREFAS DO TIPO CR.	31
TABELA 5 – ALGORITMOS E TÉCNICAS MAIS UTILIZADOS PARA CADA TIPO DE APRENDIZADO DE MÁQUINA	36
TABELA 6 - HISTÓRICO DA UTILIZAÇÃO DA IA EM JOGOS.....	37
TABELA 7 – ERRO ABSOLUTO DOS VÁRIOS ALGORITMOS DE PREDIÇÃO USADO NO TRABALHO BASEADO NAS CARACTERÍSTICAS DO TEMPO PARA O 1º NÍVEL E 1º E 2º NÍVEL USANDO PARÂMETROS AJUSTADOS. MENORES VALORES SÃO MELHORES.....	44
TABELA 8 - TABELA DE AGREGAÇÃO DOS TRABALHOS RELACIONADOS POR CAMPOS DE ATUAÇÃO.....	47
TABELA 9 - TABELA DE COMPARAÇÃO.....	48
TABELA 10 – PROXIMIDADE DAS PALAVRAS. AMOSTRA DE 5 PALAVRAS.....	52
TABELA 11 – PESOS DOS TIPOS DE TENTATIVAS DE LEITURA.....	52
TABELA 12 – PESOS DOS TIPOS DE TENTATIVAS.....	54
TABELA 13 – EXEMPLO DE DADOS DE TREINAMENTO ESSENCIAIS DE CADA TAREFA PARA O MODELO DE DADOS DA REGRESSÃO LOGÍSTICA. PARA O VETOR Y, 1 = ACERTO, 0 = ERRO.....	63
TABELA 14 – TABELA COM EXEMPLOS DE CONDIÇÕES ESPECIAIS.	65
TABELA 15 – REQUISITOS GERAIS DO JOGO.....	69
TABELA 16 – CONDIÇÕES BÁSICAS PARA TRANSIÇÃO DE FASE.....	75
TABELA 17 – NÍVEL DE DIFICULDADE DAS TAREFAS, DADA A DIFICULDADE GERADA PELO SISTEMA E A DIFICULDADE SUGERIDA PELO ESPECIALISTA. ESP: ESPECIALISTA, SIS: SISTEMA FORMULADO, DIF: DIFERENÇA ENTRE O SISTEMA E A OPINIÃO DO ESPECIALISTA.....	79
TABELA 18 – TABELA DOS RESULTADOS DA PRIMEIRA E DA SEGUNDA SIMULAÇÃO.....	82
TABELA 19 – TABELA DOS RESULTADOS CLASSIFICADOS PELOS PSICÓLOGOS PARA AS SIMULAÇÕES	82
TABELA 20 – TABELA COM AS CLASSIFICAÇÕES DO APRENDIZADO GERADAS PELA MA DE CADA PALAVRA DAS SIMULAÇÕES REALIZADAS. HA: HOUVE APRENDIZADO, SA: SEM APRENDIZADO.....	91
TABELA 21 – NÍVEL DE DIFICULDADE DAS TAREFAS GERADAS DADO A DIFICULDADE GERADA PELO SISTEMA INTELIGENTE E A TAREFA ESCOLHIDA PELOS PSICÓLOGOS AGRUPADOS POR COMPORTAMENTO DO ALUNO..	93
TABELA 22 – ARTIGOS PUBLICADOS RELACIONADOS COM ESTE PROJETO DE PESQUISA.	94
TABELA 23 – INDICAÇÃO DA PROXIMIDADE DAS PALAVRAS. NÚMERO DE PALAVRAS: 15. COMBINAÇÕES: 105. EM VALOR CRESCENTE DE PROXIMIDADE, SENDO 0 = NADA PRÓXIMAS ATÉ 1 = IDÊNTICAS.....	104
TABELA 24 – TAREFAS SIMULANDO O DÉFICIT DE APRENDIZADO DE LEITURA.	105
TABELA 25 – TAREFAS SIMULANDO O APRENDIZADO GRADATIVO.....	105
TABELA 26 – TABELA COM AS TAREFAS DO PRÉ-TESTE PARA SIMULAÇÃO DOS COMPORTAMENTOS E ANÁLISE PELO SISTEMA INTELIGENTE. A RESPOSTA DOS ACERTOS E ERROS DE CADA TAREFA ESTÃO NA TABELA 27.....	108
TABELA 27 – ACERTOS DOS COMPORTAMENTOS SIMULADOS DA TABELA 26 ANTERIOR.....	111

LISTA DE SIGLAS

ALEPP	Aprendendo a Ler e Escrever em Pequenos Passos
ALERPG	Aprendendo a Ler e Escrever RPG
AP	Aprendeu Palavra
APC	Aprendizado Consolidado
APG	Aprendizado Gradativo
API	<i>Application Programming Interface</i>
CR-MTS	<i>Constructed Response - Matching to Sample</i>
DAP	Déficit de Aprendizado
DLL	<i>Dynamic Link Library</i>
FANN	<i>Fast Artificial Neural Network</i>
FSM	<i>Finite State Machine</i>
GA	Grau de Aprendizado
GBL	<i>Game Based Learning</i>
GEIC	Gerenciador de Ensino Individualizado por Computador
HMM	<i>Hidden Markov Model</i>
IA	Inteligência artificial
IEEE	<i>Institute of Electrical and Electronic Engineers</i>
IDE	<i>Integrated Development Environment</i>
JNI	<i>Java Native Interface</i>
JPA	<i>Java Persistence API</i>
M1	Modelo de Referência 1
MA	Máquina de Aprendizado
MLP	<i>MultiLayer Perceptron</i>
MTS	<i>Matching to Sample</i>
MMORPG	<i>Massive Multiplayer Online Role-Playing Game</i>
NPC	<i>Non Player Character</i>
ONU	Organizações das nações Unidas
PCA	<i>Principal Component Analysis</i>
R1	Reforço1
RPG	<i>Role-Playing Games</i>
S+	Estímulo Positivo
S-	Estímulo Negativo
SARSA	<i>State Action Reward State Action</i>
SMO	<i>Sequential Minimal Optimization</i>
SVM	<i>Support Vector Machines</i>
UFSCar	Universidade Federal de São Carlos
UFPA	Universidade Federal do Pará
VAK	<i>Visual Auditory Kinesthetic</i>
W3C	<i>World Wide Web Consortium</i>
WEKA	<i>Waikato Environment for Knowledge Analysis</i>

LISTA DE SÍMBOLOS

m	<i>Número de Exemplos do Conjunto de dados</i>
X	<i>Matriz do Conjunto de Dados referentes a Dificuldade das Tarefas</i>
$X^{(i)}$	<i>i-ésima linha do objeto do conjunto de dados X</i>
X_j	<i>j-ésima coluna de atributos de dados do conjunto de dados X</i>
$X_j^{(i)}$	<i>Valor do j-ésimo atributo do i-ésimo objeto do conjunto de dados X</i>
y	<i>Vetor com as classes do conjunto de treino referentes ao acerto ou erro das tarefas</i>
$y^{(i)}$	<i>i-ésimo valor do vetor de classes y</i>
θ	<i>Vetor com os coeficientes de aprendizado</i>
θ_n	<i>n-ésimo valor de coeficiente do vetor θ</i>
α	<i>Taxa de aprendizado do algoritmo GradientDescent</i>
$h_{\theta}(x)$	<i>Hipótese aplicada a uma matriz de dificuldade como valor de entrada x</i>
e	<i>Número Neperiano</i>
z	<i>Valor z que representa fatores diversos</i>
$g(z)$	<i>Função sigmoid aplicada a z</i>

SUMÁRIO

1. Introdução.....	17
1.1. <i>Objetivos</i>	19
1.2. <i>Justificativa</i>	20
1.3. <i>Metodologia</i>	21
1.4. <i>Estrutura da Dissertação</i>	24
2. Estado da arte.....	26
2.1. <i>Análise comportamental no Processo de Ensino</i>	26
2.1.1. <i>Programa de Ensino ALEPP</i>	30
2.2. <i>Jogos Digitais Educacionais</i>	32
2.3. <i>Aprendizado de Máquina</i>	33
2.3.1. <i>Aprendizado de Máquina em Jogos Digitais</i>	36
2.4. <i>Trabalhos Relacionados</i>	38
2.4.1. <i>GEIC</i>	38
2.4.2. <i>ALE-RPG</i>	40
2.4.3. <i>Grapho Game</i>	41
2.4.4. <i>Darwin Kombat</i>	42
2.4.5. <i>Estudo de personagens do tipo Black and White</i>	43
2.4.6. <i>Predição do Comportamento do Jogador em Tomb Raider</i>	43
2.4.7. <i>Classificação de gamebots em um jogo digital de futebol com HMM</i>	44
2.4.8. <i>Classificação de comportamentos via Rede Neural</i>	46
2.4.9. <i>Análise sobre os Trabalhos Relacionados</i>	47
3. Modelagem da complexidade de uma tarefa de ensino.....	50
3.1. <i>Dificuldade de uma tarefa de ensino</i>	50
3.1.1. <i>Dificuldade de Leitura</i>	51
3.1.2. <i>Dificuldade de Escrita</i>	53
4. Máquina de Aprendizado aplicada no Processo de Ensino	57
4.1. <i>Aprendizado Supervisionado para o Programa de Ensino</i>	57
4.1.1. <i>Representação da Hipótese da Regressão Logística</i>	60
4.1.2. <i>Treinamento da máquina de aprendizado</i>	62
4.2. <i>Modelo de máquina de aprendizado proposto</i>	64
4.2.1. <i>Decisão do Aprendizado e parâmetros auxiliares</i>	64
4.2.2. <i>Protótipo do modelo de máquina de aprendizado</i>	67
5. Projeto de Jogo: As Aventuras de Amaru.....	69
5.1. <i>Visão Geral</i>	69
5.2. <i>Enredo e Personagens Principais</i>	71
5.3. <i>Ferramentas Tecnológicas</i>	73

5.4. <i>Lógica de Geração de Sessão de Ensino</i>	74
5.5. <i>Programa: Relatório Amaru</i>	75
6. Experimentos e Resultados	78
6.1. <i>Validação das equações de dificuldade</i>	78
6.2. <i>Experimento da análise do aprendizado da leitura</i>	80
6.3. <i>Experimento simulado sobre a análise comportamental de alunos</i>	82
6.3.1. <i>Metodologia do experimento dos comportamentos de alunos</i>	82
6.3.2. <i>Resultado e Análise dos parâmetros da Máquina de Aprendizado</i>	83
6.3.3. <i>Resultado sobre a geração de tarefas de ensino</i>	92
7. Conclusões	94
7.1. <i>Publicações</i>	94
7.2. <i>Trabalhos Futuros</i>	95
Referências Bibliográficas	96
Anexos	103
Anexo A. Tabela Proximidade das Palavras.	104
Anexo B. Tabelas de Dados para simulação da Aquisição da Leitura	105
Anexo C. Tabela de Dados para simulação de comportamentos.	107
Apêndices	113
Apêndice A. Algoritmo de Proximidade de Palavras.	114
Apêndice B. Algoritmo: Lógica de geração de sessão de ensino.	117
Apêndice C. Questionário para Validação do Aprendizado de Leitura	119
Apêndice D. Questionário para Validação da Dificuldade e Geração das Tarefas	121

1. Introdução

O Brasil tem mais de 4,5 milhões de crianças entre 5 e 9 anos que não são alfabetizadas segundo os dados do IBGE (Instituto Brasileiro de Geografia e Estatística). A taxa de reprovação entre alunos da 2ª série (3º ano) do ensino fundamental no ano de 2007 com idades entre 8 e 9 anos foi de 13,9%. No ano de 2010 a taxa de reprovação permaneceu no valor de 12,1%, uma leve queda de 1,8% segundo fontes do IBGE (IBGE, 2011). Houve uma redução de 29% de pessoas não alfabetizadas relativo ao censo do ano de 2010 em relação aos números apresentados em 2000. Apesar da melhora, esses valores são insatisfatórios para o país. Hoje, é considerada alfabetizada a pessoa capaz de ler e escrever um bilhete simples (Naoe, 2012).

Esses dados revelam a carência do ensino fundamental, que é responsável pelo desenvolvimento do domínio da leitura e escrita dos alunos. Fica claro que o processo clássico de ensino e aprendizado no Brasil necessita de auxílios com novos métodos e inovações.

É em virtude dessa deficiência na alfabetização do Brasil que o programa de ensino “Aprendendo a Ler e Escrever em Pequenos Passos” (ALEPP) (Rose & Souza *et al.*, 1989) foi criado. Os pesquisadores da Universidade Federal de São Carlos (UFSCar) utilizaram características da análise comportamental para auxiliar o ensino de alunos na aquisição da escrita e leitura com ou sem histórico de fracasso (Souza & Rose, 2006). Com o passar dos anos, o programa ganhou versões informatizadas com o objetivo de gerenciar as tarefas de ensino.

Outro meio para auxiliar na alfabetização de crianças é através de jogos digitais que estimulem o desafio e gerem motivação aos alunos, sem excluir o fator educacional. Os jogos eletrônicos podem ser usados como mais uma ferramenta de auxílio no ensino-aprendizagem (Amate & Slaets *et al.*, 2003). Observou-se que, crianças em fase de alfabetização se mostravam fascinadas por tecnologias vigentes e conseguiam assimilar com facilidade alguns termos propostos pelo computador (Aguiar, 2008).

Estudiosos da psicologia cognitiva se mostraram interessados em utilizar os jogos digitais com a possibilidade de revelar um leque para novos estudos, ao acrescentar métodos capazes de auxiliar no processo de cognição humana (Sayeg & Conselho, 2000). Um jogo de RPG (*Role Playing Game*) denominado “Aprendendo a Ler e Escrever RPG” (ALE-RPG) foi concebido com a função de tornar o ensino das tarefas de aprendizado mais divertido (Siqueira, 2012). Contudo, muitos dos estudos relacionados com a psicologia, educação e

jogos digitais não se aplicaram em utilizar uma forma computacional inteligente que buscasse classificar e prever o aprendizado do aluno de forma individualizada auxiliando o processo de análise do aprendizado.

Deve-se destacar a função da inteligência artificial (IA) para solucionar esse problema aliado aos jogos digitais. Afinal, desde a década de 80 os jogos computacionais tornaram-se um produto de grande interesse para a indústria e academia. O advento de empresas com lançamentos interativos, consoles e fliperamas trouxe considerável diversão para os jogadores da época (Kishimoto, 2004). Por outro lado, a IA voltava de uma crise que a fez diminuir o ritmo de agregação do conhecimento por anos (Russel & Norvig, 2004).

Logo os pesquisadores relacionados com a inteligência computacional buscaram alcançar o tempo perdido com várias pesquisas e técnicas desafiadoras tornando-se uma questão de tempo em aliar a IA com os jogos computacionais. Na década de 90, a indústria de *games* produziu vários títulos e a academia gerou diversas pesquisas, mas o avanço realmente começou no início do milênio, com técnicas de IA mais elaboradas que realizassem características de aprendizado computacional em jogos, chamadas de aprendizado de máquina.

Novas pesquisas em aprendizado de máquina abriram modelos e revelaram padrões com características interessantes para a indústria de jogos, visto a preocupação com uma interação mais realista, divertida e natural. Os métodos de aprendizado de máquina (em inglês *machine learning*) tendem a serem uma das mais promissoras técnicas na área de jogos, pois possibilita, entre diversos métodos, o suporte à previsão e classificação computacional.

O cenário de análise comportamental em que o ALEPP se encontra também está ligado a área de estudo de aprendizado de máquina, pois segundo (Mitchell, 2006):

“teorias e algoritmos de aprendizado foram encontrados relevantes para a compreensão de aspectos do aprendizado humano e animal (...). No entanto as teorias da aprendizagem animal envolvem considerações que ainda não foram considerados no aprendizado de máquina como, o papel de motivação, medo, urgência, esquecimento e o aprendizado sobre escala múltipla de tempo. Há uma rica oportunidade para a fecundação cruzada aqui, uma oportunidade de desenvolver uma teoria geral dos processos de aprendizagem, relativos a animais, bem como máquinas potenciais de implicação para estratégias de melhoria do ensino para estudantes”.

O desafio em analisar e identificar automaticamente o aprendizado e o comportamento do aluno baseado em um programa de ensino via aprendizado de máquina trás uma notável oportunidade de pesquisas, além de auxiliar a pedagogos e psicólogos que trabalham com o

ensino. Sistemas de inclusão de tarefas adaptadas e personalizadas para o aluno que possui ou não déficit de aprendizado podem ser utilizados baseados nas previsões e classificações geradas por técnicas de aprendizado de máquina.

Em particular, neste trabalho é proposto um modelo de aprendizado de máquina para ser utilizado em um jogo digital, a fim de analisar dados comportamentais no processo de aprendizado vindouro do ALEPP de forma personalizada. Um protótipo de máquina de aprendizado foi desenvolvido contribuindo com a avaliação do grau de aprendizagem da leitura e escrita de palavras individuais, através da classificação de palavras e previsão de acerto/erro das tarefas de ensino. Com isso, é possível identificar o comportamento do aluno durante uma sessão de estudo. Pode-se também determinar o aprendizado de cada palavra do jogo e obter informações relevantes sobre possíveis decisões no processo de ensino, tornando essa pesquisa singular para as áreas de estudo envolvidas.

1.1.Objetivos

Este trabalho tem como seu principal objetivo a elaboração de um modelo de máquina de aprendizado para auxiliar a análise comportamental de alunos através de um jogo educacional no contexto de leitura e escrita. Destaca-se que este trabalho faz parte de um projeto maior que envolve equipes multidisciplinares. Um dos objetivos secundários é projetar a máquina de aprendizado de forma simples para ser acoplada em dispositivos móveis, ampliando quantitativamente e futuramente o público em questão.

Os objetivos específicos desta dissertação são:

- Criar um protótipo de máquina de aprendizado para classificar e prever os aspectos do comportamento do aluno.
- Levar em consideração características de leitura e escrita do programa de ensino no modelo de aprendizado de máquina.
- Construir um jogo, no contexto de ensino da leitura e da escrita, que tenha um caráter lúdico baseado em métodos da análise comportamental do ensino.
- Integrar o protótipo de máquina de aprendizado com o protótipo do jogo em uma arquitetura adequada.
- Desenvolver um programa para gerar um relatório de desempenho do aluno no jogo.
- Realizar os testes necessários em conjunto com especialistas da área de psicologia.

- Projetar o jogo para ser executado tanto em dispositivos móveis quanto em computadores *desktops*.

1.2. Justificativa

Há um crescente interesse acadêmico de construir jogos digitais com fatores lúdicos que apresentem um novo auxílio para o aprendizado de alunos (Azevedo & Marques, 2001). Porém, as pesquisas de cunho a realizar o mapeamento do comportamento do aprendiz através de jogos digitais ainda são raras. Esse mapeamento de forma inteligente reflete um desafio para os estudiosos da ciência da computação que utilizam a inteligência artificial para reconhecer o comportamento do jogador.

A possibilidade de “aprendizado” do computador baseado em entradas dinâmicas enviadas pelo jogador, expressa uma solução técnica da inteligência artificial bastante utilizada: o aprendizado de máquina. Contudo, as técnicas dessa forma de aprendizado computacional com o objetivo de auxiliar no aprendizado humano provavelmente se tornariam mais eficientes se fossem baseadas em um programa fidedigno da psicologia comportamental.

Segundo Azevedo & Marques (2001), a escrita e leitura são dois repertórios inicialmente múltiplos e que envolvem uma série de analogias diferentes, ambas são tratadas como tarefas distintas e proporcionam certa dificuldade em seu aprendizado. Conforme Cardoso (2005), existe um alto índice de crianças que apresentam problemas na aprendizagem de leitura e escrita e, em virtude desses problemas de aprendizagem torna-se necessário a adição de artifícios de ensino inovadores com a meta de tentar remediar tais falhas.

Em vista da problemática descrita, a justificativa para este trabalho é obter, através de um modelo de máquina de aprendizado, informações do comportamento do jogador/aluno para auxiliar o analista comportamental com novas maneiras de analisar dados, a fim de aperfeiçoar o ensino de crianças em fase de aprendizado. Desta forma, portas para envolver o jogador com a geração de tarefas de acordo com o aprendizado do jogador podem ser abertas. Por exemplo: o jogo pode criar novas tarefas de ensino adaptadas se existir uma análise automatizada do comportamento do jogador.

O produto deste trabalho, tanto da parte técnica quanto da parte científica, tende a ser de grande importância, contribuindo para as seguintes áreas do conhecimento:

- **Educação:** este trabalho possui temática educacional no ensino da leitura e da escrita, ou seja, com a utilização de um jogo no processo de ensino-

aprendizagem. Esse atributo é relevante para usuários inseridos no contexto educacional.

- **Análise do comportamento:** este trabalho tem como referência a utilização do programa de ensino ALEPP, que é objeto de grandes estudos na área de análise comportamental (Souza & Rose, 2006). Outras pesquisas poderão ser iniciadas com a análise deste sistema de ensino.
- **Jogos digitais:** este trabalho visa a construção de um jogo digital, para servir de plataforma e objeto de aprendizagem aos alunos. Logo, técnicas ligadas ao *Game Development* foram utilizadas agregando conhecimento para o grupo de alunos formado na UFPA para o desenvolvimento do jogo.
- **Sistemas inteligentes:** este trabalho utiliza técnicas de inteligência artificial a fim de classificar e/ou prever o comportamento do jogador/aluno. Essas técnicas necessitam de ampla análise, testes e validação gerando estudos interessantes nessa área. Com a técnica implantada, o analista do comportamento poderá utilizar mais uma ferramenta como agente inteligente a seu dispor.

1.3. Metodologia

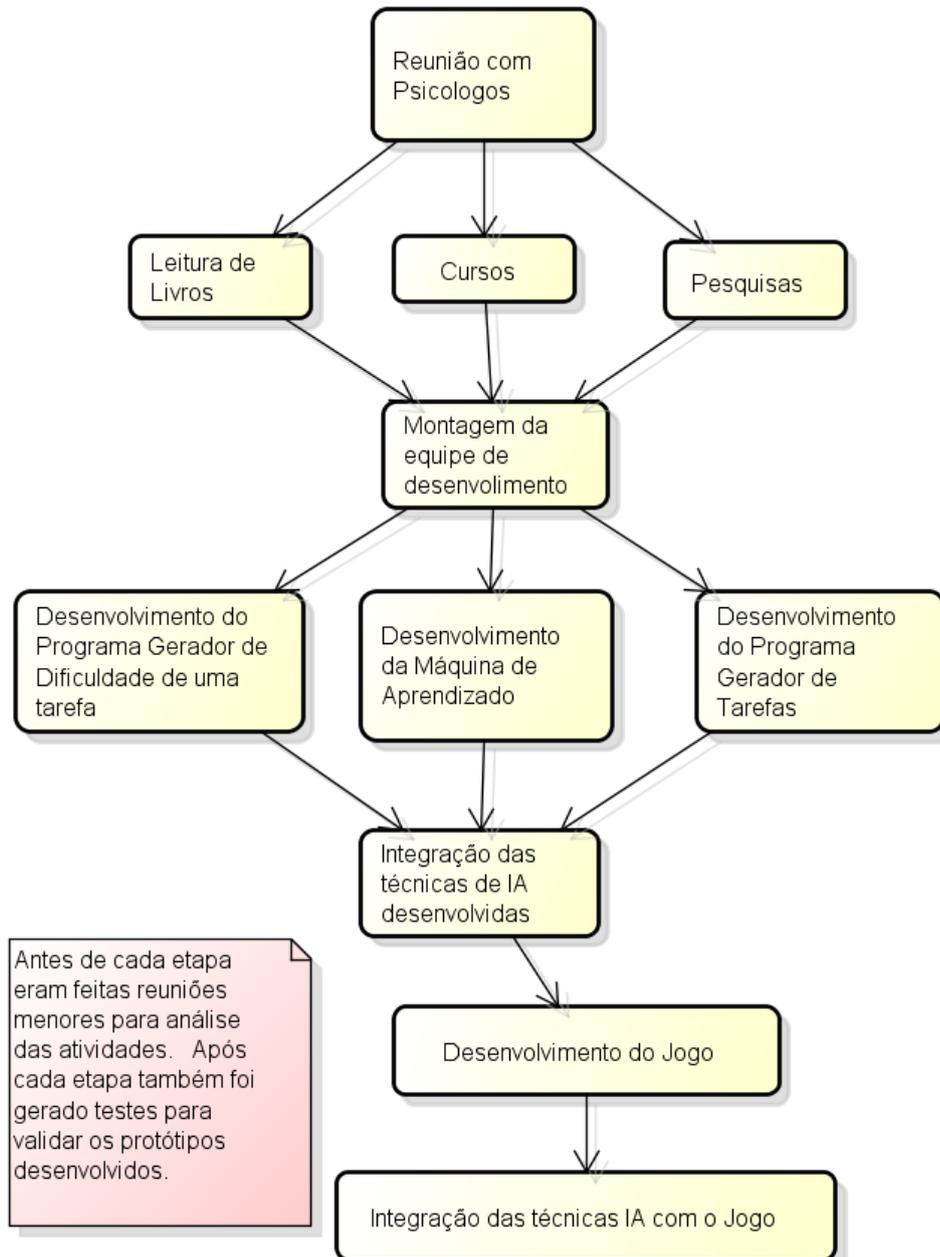
Para o desenvolvimento desta pesquisa foi realizada reuniões com os psicólogos para debater os problemas, possíveis soluções e inovações baseado na estrutura do ALEPP. Em seguida foi realizada uma revisão bibliográfica para verificar os principais trabalhos produzidos por pesquisadores reconhecidos na área de educação, psicologia, computação e de jogos digitais destinados à aprendizagem.

Artigos em periódicos e eventos científicos tiveram maior atenção, considerando os aspectos de cada área com o tema central. O aprendizado de máquina relacionado com análise comportamental e jogos digitais teve maior prioridade, mas não descartando os trabalhos com essas áreas individualmente. Outra prioridade importante foi pesquisar jogos digitais envolvidos com educação e análise comportamental, visando identificar trabalhos relacionados significativos.

No geral, várias *strings* foram inseridas em motores de busca como o IEEE Explorer (IEEE, 2012) e em congressos especializados retornando 33 artigos que, por critério de interesse, somente 10 foram categorizados de considerável importância.

Paralelamente com a pesquisa foi realizado um levantamento sobre os métodos de IA visando encontrar uma técnica adequada ao problema proposto. Isso ocorreu através de estudo em cursos, livros e artigos como consta na Figura 1.

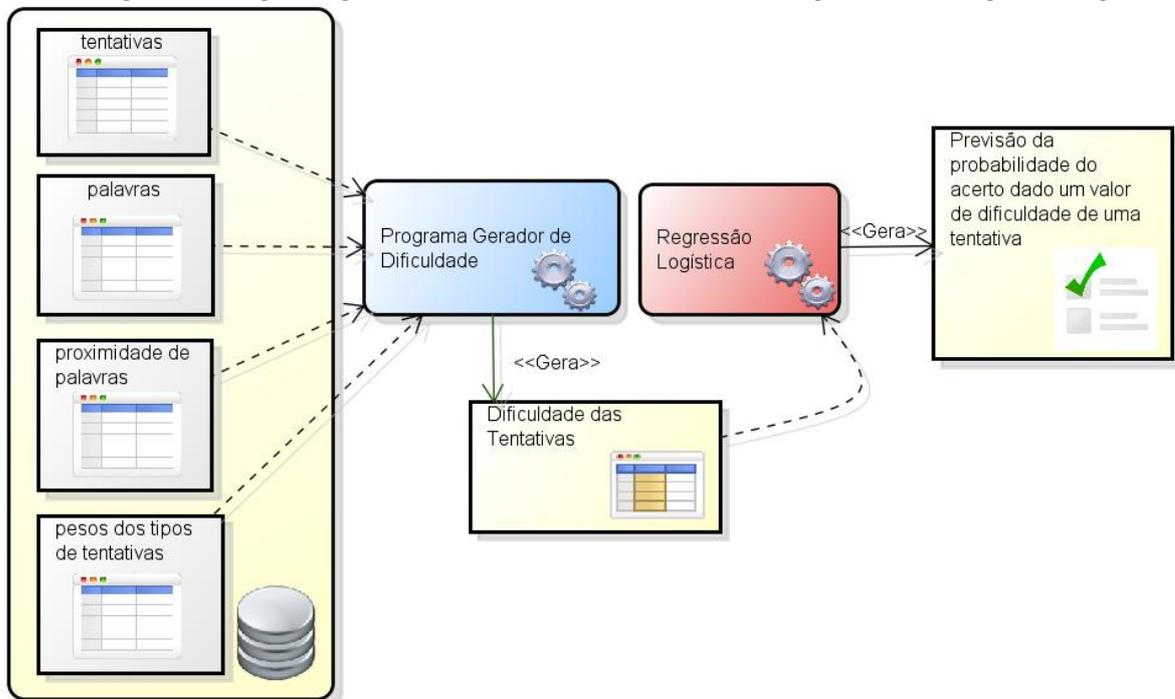
Figura 1 – Esboço da metodologia utilizada.



Determinado uma técnica e o tipo de problema de aprendizado de máquina partiu-se para a construção de um modelo, com auxílio de psicólogos, para verificar se realmente a técnica de IA poderia ser encaixada com o programa de ensino. Em seguida foi desenvolvido o primeiro protótipo de máquina de aprendizado o que gerou alguns resultados. Resumidamente, no modelo proposto foi implementado um programa que gera a dificuldade de cada tarefa de ensino baseado no banco de dados que armazena os dados de evolução do aluno. Para a técnica de regressão logística é apresentado a dificuldade de cada tarefa e se o

aluno acertou ou errou determinada tarefa. Nessa etapa do trabalho, foi integrada a **regressão logística** com o **programa gerador da dificuldade das tarefas de ensino** ilustrado na Figura 2, a fim de obter a previsão da probabilidade de acerto dado um valor de dificuldade qualquer.

Figura 2 - Programa gerador de dificuldade das tentativas integrado com a regressão logística.



O planejamento inicial foi reconfigurado para adaptar-se a questão técnica da construção do jogo com o módulo de IA construído no primeiro protótipo. Também outra técnica de IA relacionada com a geração de tarefas de aprendizado foi desenvolvida por outro pesquisador a fim de gerar tarefas de ensino personalizadas através de um sistema *fuzzy* (Pereira, 2013).

Uma equipe foi formada o que possibilitou a criação de protótipos de jogo funcionais e de programas de IA. Na Tabela 1 é mostrado os papéis e participantes do projeto geral.

Tabela 1 – Participantes do Projeto.

Participante	Papel
Gilberto Nerino	Pesquisador e Programador de Técnicas de IA
Adalberto Bosco	Pesquisador e Programador de Técnicas de IA
Pedro Afonso	Programador de Jogos Digitais
Hyago Pinheiro	Programador de Jogos Digitais
Rafael Bessa	Artista
Ellton Sales	Programador
Leonardo Marques	Psicólogo e Analista Comportamental
Dionne Monteiro	Orientador

Os testes e resultados foram validados através de questionários com analistas do comportamento da Universidade Federal de São Carlos sobre simulações induzindo comportamentos frequentes de alunos. Isso possibilitou uma avaliação de especialistas da máquina de aprendizado para o ambiente formulado.

1.4.Estrutura da Dissertação

Este trabalho é composto por sete capítulos, incluindo anexos e apêndices. Além deste capítulo introdutório, esta dissertação está dividida em:

Capítulo 2: apresenta os principais conceitos sobre: análise comportamental, o programa de ensino adotado, jogos digitais na educação e conceitos de aprendizado de máquina. No final deste capítulo, é feita uma análise de todos os trabalhos relacionados mostrando uma relação e agrupando cada trabalho e sua contribuição com as linhas de conhecimento citadas.

Capítulo 3: trata da complexidade das tarefas de ensino buscando identificar os fatores de dificuldade implícitos na estrutura do programa de ensino através de equações de dificuldade.

Capítulo 4: é apresentado o projeto da máquina de aprendizado e como foi correlacionado características do programa de ensino junto à técnica de *regressão logística*, determinada para a resolução do problema de aprendizado supervisionado.

Capítulo 5: são mostradas características de enredo, personagens, cenário, as ferramentas utilizadas, metodologia de desenvolvimento usada e a estrutura geral do protótipo de jogo chamado “As aventuras de Amaru”.

Capítulo 6: são apresentados os resultados realizados da máquina de aprendizado, junto com análise dos dados obtidos nos testes.

Capítulo 7: são apresentadas às considerações finais sobre o trabalho e propostas para trabalhos futuros.

Anexo A: Apresenta uma tabela de dados produzidos pelos psicólogos para identificar o quanto uma palavra é próxima da outra em valores percentuais no programa de ensino.

Anexo B: Apresenta os dados das simulações utilizadas para aquisição de leitura no protótipo de máquina de aprendizado.

Anexo C: Apresenta as tabelas de dados utilizadas como dados de treino para o experimento com as simulações dos comportamentos mais frequentes dos alunos.

Apêndice A: Define o algoritmo de proximidade de palavras em pseudocódigo.

Apêndice B: Define o algoritmo de lógica de geração de sessão de ensino que gera uma sessão de ensino para o aluno.

Apêndice C: Apresenta o modelo de questionário utilizado para validação dos dados do protótipo de máquina de aprendizado para a aquisição de leitura.

Apêndice D: Apresenta o questionário utilizado para validar os dados referentes a dificuldade das tarefas de ensino e as simulações dos comportamentos dos alunos.

2. Estado da arte

Neste capítulo serão apresentados os principais conceitos do estudo sobre o estado da arte no que diz respeito à análise comportamental no programa de ensino utilizado e modelos de aprendizado de máquina aliados a jogos educacionais apresentando como os mesmos têm papel e espaço no processo de aprendizagem. Serão abordados tanto trabalhos que possuem a finalidade de ensinar como outros que utilizaram a aprendizado de máquina para os mais diferentes conteúdos. No final deste capítulo são apresentadas informações que comparam os trabalhos pesquisados.

2.1. Análise comportamental no Processo de Ensino

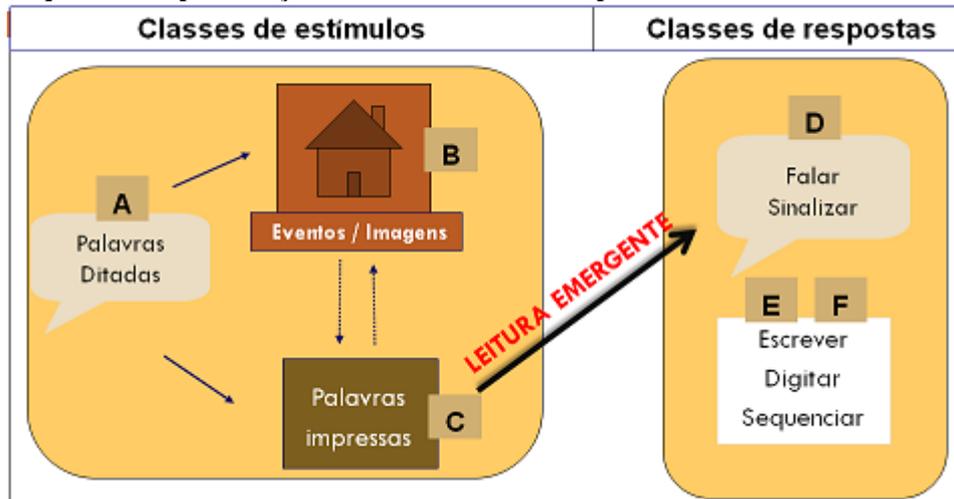
A análise comportamental está ligada com a **psicologia experimental cognitiva** e visa ser uma ferramenta para o objeto de estudo da psicologia (Todorov, 1982). Observar como uma criança se comporta no procedimento de aprendizado e delinear características importantes para o ensino pode favorecer excepcionalmente no processo de ensino.

Agregado aos vários conceitos da análise comportamental deve-se denotar o conceito de **condicionamento operante**, que adiciona diversos aspectos, dentre eles as características de um modelo de referência, comparações com o modelo de referência, ações do indivíduo e reforços que podem ser reforços positivos, negativos ou neutros (Galvão & Barros, 2001).

Segundo Mizukami (1986), o condicionamento operante é um: *“comportamento voluntário e abrange uma quantidade muito maior da atividade humana - desde os comportamentos do bebê de balbuciar, agarrar objetos, olhar os enfeites do berço até os comportamentos mais sofisticados que o adulto apresenta”*.

O comportamento operante utiliza, dentro de seu escopo, a **teoria da equivalência de estímulos**. Essa teoria estabelece critérios de relações simbólicas (estímulos). Essa teoria pode ser simulada em laboratório (Sidman & Bortoloti e De Rose, 2007). Em termos estruturais e de forma simplificada os estímulos são considerados **imagens, sons** ou **textos**. Podemos visualizar a relação dos estímulos, para o contexto de educação de escrita e leitura, de forma mais coerente na figura abaixo.

Figura 3 – Esquema de representação da classe estímulos e respostas em conformidade com suas relações



As classes de estímulos são os objetos de apresentação para um indivíduo. Sabe-se que há uma relação entre estímulos. A figura “Casa” tem relação com a palavra “CASA” que por sua vez tem relação com o som da palavra ditada “Casa”. São relações comuns que o indivíduo em fase de aprendizado necessita afeiçoar-se, ou seja, aprender a identificar a relação entre esses estímulos.

As classes das respostas são os retornos ativos que o indivíduo produz na apresentação desses estímulos. Ao ver o estímulo “Casa” em forma de figura, som ou texto, o indivíduo no processo de ensino, atua com uma resposta: Falar, Sinalizar, Escrever, Digitar ou Sequenciar. Essa relação se configura na aquisição da **leitura emergente**. Os psicólogos identificam, para fins práticos, os estímulos como letras. A Tabela 2 simplifica a representação.

Tabela 2 – Representação de tipo de estímulos identificado como letras.

Representação do tipo do Estímulo	Tipo de Estímulo
A	Palavra Ditada. Por exemplo: pode ser ditada por um professor ou por meios tecnológicos como um computador através de sons emitidos.
B	Figura.
C	Palavra escrita. Por exemplo: é uma palavra escrita textualmente e apresentada por uma tela de computador ou um quadro negro.
D	Falar ou Sinalizar. É uma resposta dada pelo indivíduo. Significa a fala em voz clara identificando o texto, som ou figura. Pode também apontar ou selecionar através do mouse o Estímulo em questão.
E	Escrever. O indivíduo digita, escreve ou compõem o Estímulo que foi apresentado a ele.

Os analistas comportamentais identificam o aprendizado através da relação entre esses estímulos. Apresentar o “estímulo-figura” com o “estímulo-texto” e obter uma resposta correta indica, salvo algumas outras ponderações, que o indivíduo está aprendendo a identificar a figura e o texto corretamente. Em um próximo passo, por exemplo, se ele consegue escutar uma palavra e a escrever logo em seguida, significa que o processo de aprendizado de escutar e escrever estão avançando no **programa de ensino**.

A teoria da **programação do ensino** ou **programa de ensino** é, conforme Juliano (2009):

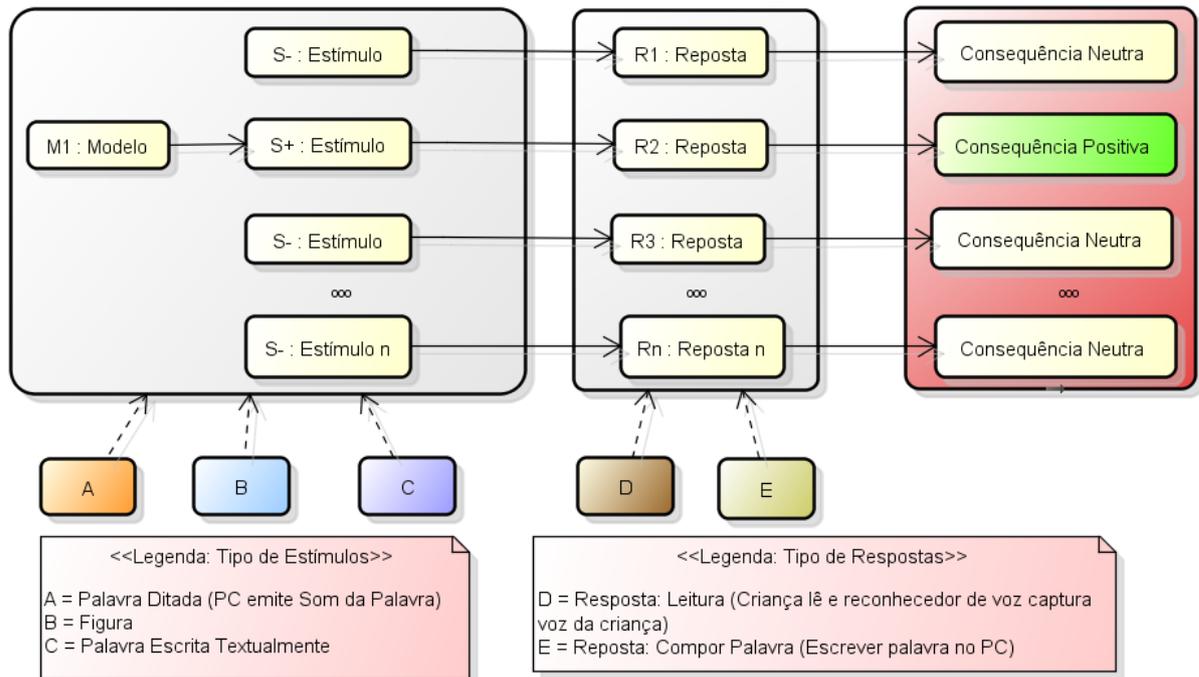
“um arranjo de contingências que consiste na elaboração das respostas finais que se espera do estudante em termos de objetivos comportamentais e, com base nessas respostas, organiza-se um conjunto de materiais (estímulos) com uma ordem crescente de dificuldade ou complexidade, considerando o repertório atual do estudante e visando a aumentar gradualmente esse repertório no sentido da consecução dos objetivos comportamentais propostos por meio do reforçamento positivo”.

Os estímulos são configurados em uma **tentativa de ensino** que é uma forma de **tarefa de ensino** oferecida ao indivíduo que está passando pelo processo de instrução (Sidman & Tailby, 2006). A Figura 4 representa o fluxo de uma tentativa de ensino onde:

- M1 significa o Modelo de referência que é uma dica auxiliar para o indivíduo;
- S+ significa o estímulo positivo, ou estímulo correto;
- S- o estímulo negativo ou estímulo incorreto;
- R1, R2 até Rn significam as respostas que se pode obter por parte do aprendiz;
- a consequência positiva significa um tipo de reforço positivo ou seja um estímulo de incentivo identificando que o indivíduo está no caminho certo;
- a consequência neutra significa a ausência de reforço sendo ele positivo ou negativo.

Verifica-se também que há os tipos dos estímulos nesse fluxo que agregam os tipos **A, B, C, D** e **E** para cada estímulo.

Figura 4 – Fluxo da discriminação condicional no programa de ensino



A aprendizagem complexa (Skinner, 1950) e a cognição utilizam de diversos conceitos onde a **discriminação condicional** é um desses conceitos importantes para os elementos de ensino. Pode-se considerar a definição de uma relação de discriminação condicional conforme Debert *et al.* (2006):

“uma situação de **discriminação condicional** não se estabelece uma relação constante entre um estímulo antecedente e uma resposta; esta relação muda de acordo com os contextos nos quais este estímulo aparece. Em linhas gerais, para se obter uma relação condicional, deve-se reforçar determinada resposta na presença de um estímulo específico apenas se um outro estímulo estiver presente. Apenas na presença desta combinação de dois estímulos, as respostas são seguidas de reforço. Combinações outras desses estímulos com outros estímulos não se configuram ocasiões diante das quais as respostas são seguidas de reforço, mesmo se estas combinações compartilharem um dos estímulos com as combinações diante das quais as respostas foram seguidas de reforço”.

Com esse conceito pode-se criar uma tarefa ou tentativa de forma bem estruturada envolvendo estímulos, respostas da criança, reforços positivos caso ela esteja progredindo ou reforços neutros se ela não está tendo um bom andamento no exercício. Os estímulos podem ser A, B ou C e devem ser organizados com um estímulo Modelo e um conjunto de estímulos de Comparações ou Escolhas onde dentre esses estímulos de comparações um deles é igual ao estímulo modelo (não necessariamente o mesmo tipo de estímulo). A resposta pode ser do tipo D ou E. Caso a criança responda corretamente ela leva uma consequência positiva. Senão

leva uma consequência neutra. Não é aconselhável aplicar uma consequência negativa ou punitiva, pois poderia frustrar a criança no processo de aprendizado.

2.1.1. Programa de Ensino ALEPP

Antes de começar a descrever outros aspectos dos tipos de relações entre tarefas, primeiramente é conveniente conhecer o programa de ensino que carrega todo o conteúdo psicoeducativo utilizado nesse trabalho.

O programa “Aprendendo a Ler e a Escrever em Pequenos Passos” (ALEPP) (Rose & Souza *et al.*, 1989) instrui pessoas que possuem dificuldades em leitura e escrita através de características planejadas. Uma delas é o fato do programa ser individualizado, o que permite que a pessoa cumpra o fluxo de ensino conforme seu ritmo de aprendizado. A divisão do repertório de ensino em unidades de ensino é outra vantagem visto que o aprendiz pode repeti-las caso tenha dificuldade em algum momento no seu progresso. Inicialmente o ALEPP era reproduzido por cartolinas, fotos, objetos, e somente mais recentemente por meios informatizados.

Dentre os tipos de relações incluídas no ALEPP destaca-se o procedimento de emparelhamento ao modelo de referência, também conhecido como *matching-to-sample* (MTS) (Debert & Matos e Andery, 2006). Neste procedimento, muito comum na aquisição de leitura, o participante deve escolher uma entre duas ou mais alternativas equivalente ao modelo de referência. Os tipos de relações desse procedimento estão simplificados na Tabela 3.

Tabela 3 – Exemplos de diferentes relações entre tarefas do tipo MTS.

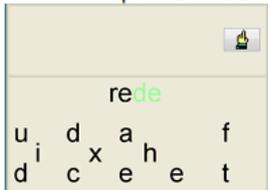
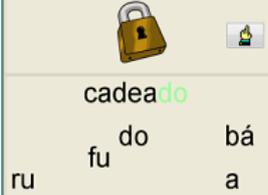
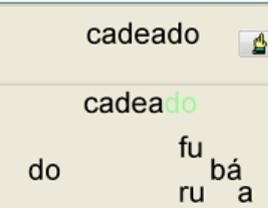
Tipo de Relação entre tarefas MTS	Ilustração da tarefa	Função do Aluno
A → B	 	Diante da instrução falada “Aponte aluno”, o aluno deverá selecionar a figura do ‘aluno’.
A → C	 	Diante da instrução falada “Aponte gato”, o aluno deverá selecionar a palavra escrita «gato».

B→C		Diante da figura de uma rede, o aluno deverá selecionar a palavra escrita «rede».
	rede remo	
C→B	vela 	Diante da palavra escrita «vela», o aluno deverá selecionar a «vela».

Fonte: Adaptado de (Marques & Golfeto e De Melo, 2011).

Outro procedimento incluso nesse programa é o CR (*constructed-response*) (Siqueira, 2012) ou construção da resposta, onde se apresenta um estímulo modelo e os diferentes estímulos de escolha. Ele consiste na escolha do estímulo de forma que se construa o modelo, parte por parte, até a estrutura se tornar equivalente ao modelo completo. Esse procedimento é muito utilizado para montar palavras letra por letra ou sílaba por sílaba até apresentar a palavra totalmente construída e equivalente com a palavra de referência. Os tipos de relações entre tarefas do procedimento CR se encontram na Tabela 4.

Tabela 4 – Três exemplos de diferentes relações entre tarefas do tipo CR.

Tipo de Relação entre tarefas CR	Ilustração da tarefa	Função do Aluno
A→E	 Aponte: rede 	Diante da instrução falada “Escreva rede”, o aluno deverá compor a palavra «rede», escolhendo as letras na ordem correta.
B→E	 	Diante da instrução falada “Que figura é essa?”, o aluno deverá compor a palavra «cadeado», escolhendo as letras na ordem correta.
C→E		Diante da instrução falada “Que palavra é essa?”, o aluno deverá compor a palavra «cadeado», escolhendo as sílabas na ordem correta.

Fonte: Adaptado de (Marques & Golfeto e De Melo, 2011).

2.2. Jogos Digitais Educacionais

Os jogos educacionais são jogos que tem por objetivo auxiliar no processo de aprendizagem e proporcionam métodos de estimulação no desenvolvimento cognitivo do aprendiz, com auxílio e criação de estratégias de entendimento para a solução de diversos problemas (Aguiar, 2008). São inúmeros os jogos educacionais e a variedade de aspectos que podem ser explorados por eles multiplicam a aquisição de formas cognitivas de ensino. Como exemplo, pode-se citar: palavra cruzada (jogo de construção), caça-palavra (jogo de exercício) e xadrez (jogo de raciocínio-prático). Existem também jogos que exploram a aplicação de regras, a localização, a destreza, a força, a concentração e auxiliam na ludicidade e no desenvolvimento de habilidades funcionais como, por exemplo, o popular jogo de queimada.

Já um jogo digital é uma atividade lúdica formada por ações e decisões, limitadas por regras e contexto próprio, que resultam em uma condição final executada por um software, (Prado & Dornhofer *et al.*, 2010). Esse contexto próprio pode ser encarado como o universo do jogo, com narrativa, ambiente e cenário a qual o personagem principal, estará inserido. As regras definem os desafios, o que pode ou não ser acionado, e permite que o jogo tenha um grau de dificuldade, obrigando o jogador a esforçar-se para cumprir determinadas metas. São diversas as classificações encontradas para os jogos digitais, tanto em revistas especializadas da indústria de games quanto no meio acadêmico. Podemos citar os gêneros de: estratégia, simuladores, aventura, *puzzle*, ação, luta, esporte, RPG, educacionais, etc.

Com o advento dos jogos digitais no começo da década de 80 observou-se que os jogos digitais poderiam ser usados como objetos de aprendizado bastando adaptá-los as características de aprendizado. Também chamados de *Game Based Learning*, conforme (Amate & Slaets e Heloisa, 2003), o uso de games para treinar, aprender e executar atividades em ambientes realísticos melhora o desempenho dos aprendizes, o que possibilita experiências práticas produzidas individualmente de acordo com seu estilo de aprendizagem e desempenho.

Há algumas características definidas por Prensky (2006) que tornam os jogos digitais mais atraentes como:

- Possuem uma forma de diversão o que proporciona alegria e prazer.
- Estimulam a resolução de conflitos, desafios, problemas e competições agindo na liberação de adrenalina e provocando a criatividade.
- Geralmente possuem histórias, oferecendo alguma emoção.

O aprendizado em jogos considera a utilizações de técnicas que são aplicadas por meio da aprendizagem interativa (Prensky, 2006), onde uma delas é a repetição de uma condição básica para o aprendizado. Essa técnica é conhecida como **Prática e Feedback**. Porém os projetistas de um jogo devem prever que o jogador retorne aos desafios várias vezes. Jogos muito fáceis ou usados de forma ineficaz são descartados rapidamente pelos jogadores. Jogos muito difíceis não provocam diversão para jogadores casuais sendo igualmente descartados.

Outra técnica utilizada em jogos para o ensino é o aprendizado **multi-sensorial**, que utiliza os sentidos do corpo humano (visão, audição, tato, paladar, olfato) para interagir com o ambiente do jogo, como por exemplo, falar nomes através de estímulos sonoros ou tocar em materiais diversos identificando algum objeto no jogo. Já a técnica de **Tutoria Inteligente** auxilia o jogador no seu percurso através de feedbacks contínuos enviado pelo aprendiz.

Assim, para atrair o aluno, o jogo deve ser lúdico, ou seja, deve ensinar e divertir ao mesmo tempo, incorporando a diversão para estimular a aprendizagem de conteúdos e habilidades por meio do entretenimento. Com isso mantém-se a atenção do aluno e desperta sua curiosidade para os conceitos a serem aprendidos.

Logo, segundo Siqueira *et al* (2011), é incontestável, o uso de jogos digitais como material didático o que propicia motivação, criatividade, a utilização de várias linguagens e várias representações geométricas, fazendo com que o aprendiz construa o conhecimento de forma lúdica.

2.3. Aprendizado de Máquina

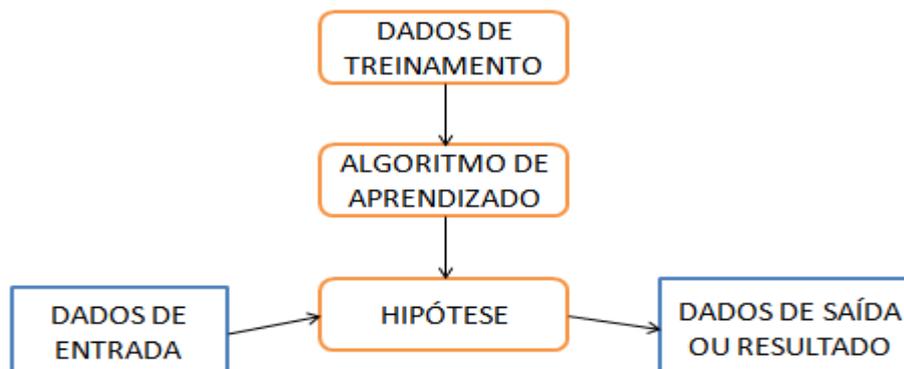
O aprendizado de máquina pode ser definido como a habilidade do computador aprender sem ser explicitamente programado (Arthur, 2011). Uma máquina de aprendizado deve se adaptar as novas circunstâncias a fim de detectar e superar padrões. Pode-se mencionar o emprego do aprendizado de máquina em diversas áreas e em aplicações como: recomendação de produtos na internet, engenharia de automação de veículos e processamento de linguagem natural.

Outro conceito recorrente para definir o aprendizado de máquina é o de Mitchell (1997) onde “*um programa de computador aprende a partir da Experiência E, em relação a uma classe de tarefas T, com medida de desempenho P, se seu desempenho em T, medido por P, melhora com E*”.

O aprendizado de máquina utiliza o conceito de **inferência indutiva**, que é o ato de derivar conclusões a partir do conhecimento e de evidências disponíveis (Mitchell, 1997). Em suma, há uma observância do fenômeno. Em seguida, ocorre a construção de um modelo

daquele fenômeno e posteriormente é possível prever informações baseado no modelo construído. A experiência pode ser fornecida pelos chamados dados de treinamento ou conjunto de dados. A Figura 5 representa uma máquina de aprendizado com seus principais elementos.

Figura 5 – Representação em alto nível de uma máquina de aprendizado



A linha de aprendizado de máquina pode ser dividida em três problemas típicos: o aprendizado supervisionado, o aprendizado não supervisionado e o aprendizado por reforço (Russel & Norvig, 2004).

O **Aprendizado supervisionado** (possui algum tipo de “supervisor”) refere-se ao fato de envolver uma função clara a partir dos dados de entradas e saídas, onde o agente inteligente pode se basear em amostras corretas de dados. O objetivo é encontrar uma função para prever um valor para novos dados.

Os métodos do aprendizado supervisionado se distinguem em dois tipos, como consta na Figura 6.

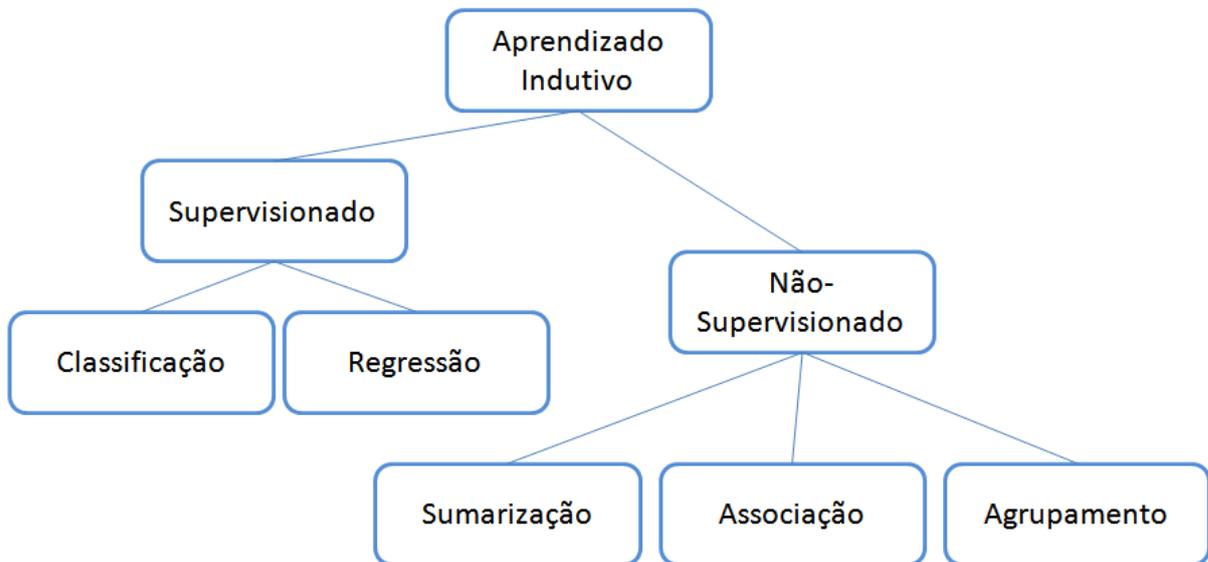
- **Classificação:** usado para rótulos de dados discretos. Por exemplo: um diagnóstico para classificar se um paciente está ou não infectado com uma doença específica.
- **Regressão:** utilizado para rótulos de dados contínuos. Por exemplo: a previsão do valor monetário de uma ação de seguro baseado em dados de compra e venda anteriores.

Aprendizado não supervisionado (não possui nenhum tipo de supervisor), antagônico ao anterior, é uma aprendizagem de padrões da entrada, quando não são fornecidos valores de saídas específicos, onde os valores ou dados são “embaralhados” sem padrões de amostras. O objetivo é explorar um conjunto de dados para encontrar grupos semelhantes.

Os métodos do aprendizado não supervisionado, como consta na Figura 6, podem ser divididos em:

- **Sumarização:** busca encontrar uma similaridade compacta entre os dados. Essa técnica é bastante utilizada para a visualização da informação e a análise e geração de relatórios.
- **Associação:** onde há padrões frequentes e associados entre as características dos dados. Dessa forma pode se admitir a relação de informações triviais e não triviais entre as relações dos atributos.
- **Agrupamento:** onde há dados agrupados de acordo com uma semelhança. Dessa forma pode se identificar categorias e grupos distintos no conjunto de dados.

Figura 6 – Visão hierárquica do aprendizado indutivo



No **Aprendizado por reforço** (“me recompense se tiver certo, me castigue se tiver errado”) em vez de ser informado sobre o que se deve fazer, induzido por um instrutor, o agente inteligente aprende através de recompensas. A falta dessa recompensa fornece ao agente uma indicação que o seu comportamento não é o aconselhável. Em casos que se deseja punir ações do agente inteligente, apresenta-se “punições” onde muitas vezes se encontra a presença de um agente crítico externo identificando as “recompensas” e os “punições”.

Para cada problema de aprendizado os cientistas e pesquisadores definiram várias técnicas computacionais que propusessem solucionar esses tipos de problemas. Aliás, existem inúmeros algoritmos e variações de algoritmos já definidos pela literatura. Na Tabela 5, encontra-se alguns dos algoritmos e técnicas mais utilizados para solucionar tais demandas.

Tabela 5 – Algoritmos e técnicas mais utilizados para cada tipo de aprendizado de máquina

Algoritmos/técnicas de aprendizado supervisionado	Algoritmos/técnicas de aprendizado não supervisionado	Algoritmos /técnicas de aprendizado por reforço
<ul style="list-style-type: none"> • Regressão Linear, • Regressão Logística, • Redes Neurais, • SVM. 	<ul style="list-style-type: none"> • <i>K-means</i>, • PCA, • <i>Anomaly detection</i>. 	<ul style="list-style-type: none"> • <i>Q-learning</i>, • SARSA.

2.3.1. Aprendizado de Máquina em Jogos Digitais

No decorrer da história dos jogos digitais, a Inteligência Artificial propôs implementar uma base sólida de algoritmos antes da chegada de técnicas que buscassem o “aprendizado” em jogos computacionais. Essa simulação de aprendizado foi constatada no meio da década de 90 no mercado de jogos, mas muito antes utilizada em pesquisas acadêmicas com o foco em novas ideias.

Alguns autores como Russel & Norvig (2004), destacam que um algoritmo de IA que não aprende ou se adapta é considerado uma **IA fraca**. Para jogos digitais, que buscam o entretenimento do jogador (Flausino, 2007), um algoritmo que simula um comportamento “inteligente” é considerado um algoritmo de IA, independente se for um algoritmo com um conjunto de condicionais ou uma rede neural complexa. Por exemplo: a ação de um NPC (*Non-Player Character*), que verifica que seu inimigo possui um armamento pesado enquanto que seu armamento é de baixo calibre, então, comparando esses dois atributos, se ver forçado a fugir do seu inimigo. Com base nesse exemplo, algumas estruturas condicionais são o suficiente para executar tal ação, mas uma estrutura mais complexa de aprendizado de máquina pode tornar a ação mais próxima do real. Ambas são programadas para aumentar a **imersão** do jogador, ou seja, a capacidade de um sistema trazer seus usuários para dentro da realidade virtual por ele construída (Ferreira, 2012). A Tabela 6 exibe uma pequena amostra da evolução da IA em jogos desde os primeiros jogos até a sua utilização em jogos criados no início do milênio.

Tabela 6 - Histórico da utilização da IA em jogos

Nível de IA	Aplicação	Ano
Sem IA	<i>Space War!</i> – primeiro jogo para computador escrito para o minicomputador PDP-1. Requer 2 jogadores	1962
	<i>Pong</i> – Versão eletrônica do tênis de mesa	1972
	Inimigos começam a aparecer	1974
Padrões	<i>Space Invaders</i> - Inimigos são padronizados, mas atiram de volta. Considerado o primeiro “clássico” com níveis, <i>score</i> , controle simples e dificuldade crescente no decorrer do jogo	1978
	<i>Pac-Man</i> – Fantasmas com movimento padronizado, mas cada fantasma possui uma personalidade	1980
	Um microcomputador vence um jogador profissional de xadrez pela primeira vez.	1983
	<i>Karate Cham (Data East, 1984)</i> – um dos primeiros jogos de luta um contra um, com o computador como adversário	1984
FSM's	<i>Herzog zweio (Technosoft, 1990)</i> – O primeiro RTS a surgir com uma implementação limitada do algoritmo de <i>pathfinding</i> .	1990
	<i>Doom (id Software, 1993)</i> – início oficial da era dos FPS (<i>First Person Shooter</i>)	1993
Varias técnicas	<i>BattleCruiser: 3000AD (Take Two Software, 1996)</i> – Primeiro uso de redes neurais em um jogo comercial	1996
	<i>Deep Blue</i> – Derrota o atual campeão de xadrez Gary Kasparov	1997
	<i>Half Life</i> – A inteligência artificial em jogos encontra-se em seu auge. O jogo faz grande uso de linguagens de <i>script</i>	1998
	<i>Black & White (Lionhead Studios, 2001)</i> – Utiliza criaturas que usam aprendizado por reforço e observação	2001

Fonte: Adaptado de (Filho, 2005).

Atualmente destacam-se novas pesquisas que estão abrindo modelos e revelando padrões com características interessantes para a indústria de jogos (Gulgemin, 2011). Uma subárea da disciplina de IA que tende como ser uma das mais promissoras na área de jogos de digitais é o aprendizado de máquina.

Russel e Norvig (2004) destacam que uma **IA forte** possui um algoritmo inteligente que obtém a competência de adaptar-se e aprender com o transcorrer do tempo. A principal característica do aprendizado de máquina em jogos computacionais é a capacidade de elementos do jogo se ajustarem a novas amostras sem serem estaticamente programados (Andrew, 2011). Em jogos computacionais, o aprendizado de máquina pode ser um aliado em diversas características como na criação de enredo, disputa contra o jogador, uma ajuda fornecida ao jogador ou até mesmo um ambiente de níveis que se adaptam à condição do jogador. Por exemplo, um NPC se tornaria um agente realmente inteligente se ele adquirisse conhecimento com o jogador e suas entradas no jogo. Outro exemplo do uso de aprendizado de máquina em jogos seria a alteração de partes da história do jogo a partir da evolução do jogador. Ou seja, os próximos níveis podem torna-se mais difíceis ou mais fáceis dependendo

do desempenho do personagem principal. Muitos jogos têm essa característica onde se destacam vários gêneros. Jogos de esportes, corrida, tiro em primeira pessoa, aventura, RPG e estratégia são os que mais utilizam as máquinas de aprendizado.

2.4.Trabalhos Relacionados

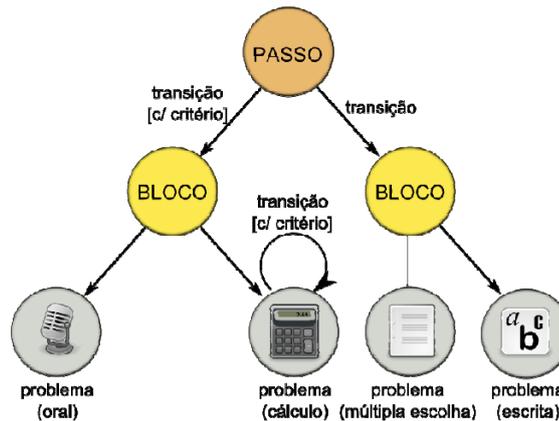
Existem diversos trabalhos relacionados com os três campos citados anteriormente: análise comportamental, jogos digitais educativos e aprendizado de máquina. Para esta pesquisa utilizou-se como referência trabalhos que foram publicados em revistas, periódicos e outros meios de pesquisa com importância científica. Foram pesquisados, primeiramente, trabalhos em cada campo individualmente e em seguida foram pesquisados trabalhos que incluíssem intercessão em duas ou nas três áreas citadas, ou seja, trabalhos mais importantes com o tema central.

2.4.1. GEIC

O Gerenciador de Ensino Individualizado por Computador (GEIC) (Orlando, 2009) é um aplicativo remoto com acesso a serviços web que possibilita a implantação do programa de ensino ALEPP. O GEIC foi desenvolvido em Java utilizando *Enterprise JavaBeans* com o *Framework JPA* e com o suporte ao servidor *GlassFish*.

O programa possui diversos módulos que permitem o gerenciamento de alunos, gerenciamento de equipe, execução da sessão de ensino, consulta em base de dados, relatórios, resultados e autoria. O módulo de Autoria, um dos módulos principais, é estruturado em “sessões de ensino”. As mesmas sessões de ensino são consideradas “passos de ensino” no qual agrega os “blocos de ensino”. Os “blocos de ensino” detêm uma série de “tarefas de ensino” que também possuem os “estímulos de ensino”. A Figura 7 simplifica essa estrutura geral.

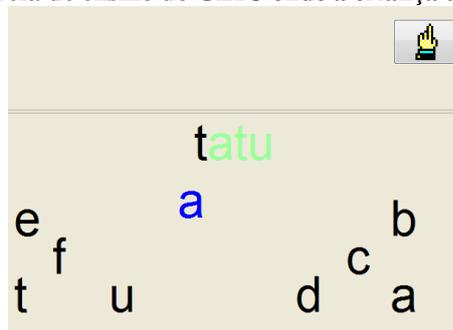
Figura 7 – Estrutura de uma Sessão de Ensino



Fonte: Orlando et al., 2009.

Desta forma o sistema foi concebido para implantar um currículo de aprendizado para leitura e escrita. Contudo, possui liberdade para configuração de outros formatos de ensino de outras disciplinas como matemática e linguagem musical, por exemplo. O programa também pretende auxiliar na reabilitação de indivíduos com problemas auditivos. A Figura 8 mostra um exemplo de uma tarefa de ensino do GEIC.

Figura 8 – Exemplo de uma tarefa de ensino do GEIC onde a criança tem que montar a palavra “tatu”



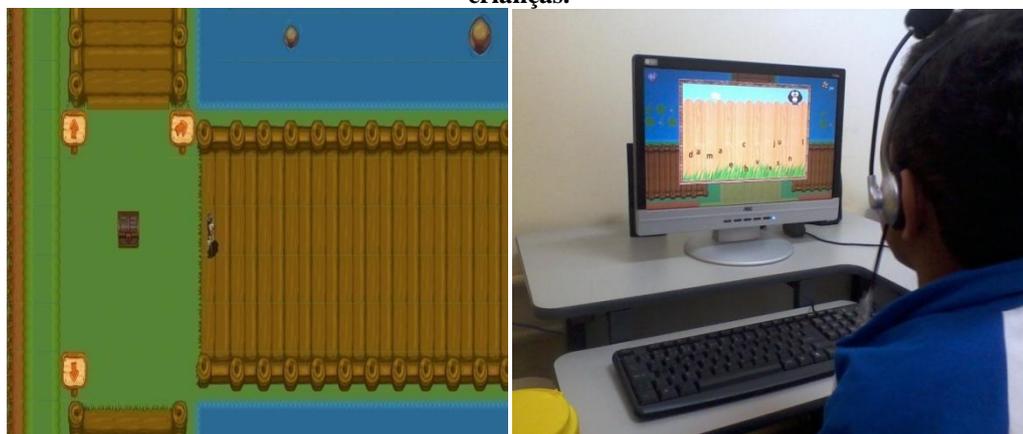
Fonte: Portela & Kampff, 2000.

O programa GEIC foi utilizado em algumas cidades do interior do estado de São Paulo, como Araraquara, Ribeirão Bonito e São Carlos. Nessas cidades, o programa foi empregado em escolas com alunos com e sem problemas de leitura e escrita, em que foram realizadas algumas sessões de ensino com duração de cerca de 25 minutos para cada aluno. No decorrer das sessões de ensino, verificou-se que algumas crianças perdiam a motivação em realizar as tarefas de ensino, uma vez que estas são realizadas de forma sequencial, ininterrupta e provocando desinteresse por parte de algumas crianças segundo Siqueira (2011).

2.4.2. ALE-RPG

O trabalho realizado por Siqueira (2012) propôs um jogo para auxiliar no ensino de crianças com dificuldades de leitura e escrita. O autor buscou uma arquitetura focada em diversas tecnologias para servir de incentivo ao aluno. Uma delas foi o reconhecimento de voz através de softwares e hardware adequados para realização de tarefas utilizadas na prática da leitura. Outra tecnologia utilizada foi a do sensor de movimentos *Kinect* da Microsoft que foi empregado para manipulação de escolhas de tarefas de construção de palavras ou indicação de figuras corretas através do movimento corporal do aluno. A Figura 9 mostra uma criança interagindo com o jogo.

Figura 9 – Uma cena do jogo ALE-RPG desenvolvido para auxiliar no aprendizado da leitura e escrita de crianças.



Fonte: Adaptado de (Siqueira, 2012).

O jogo usou uma temática de RPG com enredo próprio para atrair a atenção do aluno o que buscou, dessa forma, tornar o jogo mais lúdico. Nas tarefas e seções de aprendizado o jogo usou conceitos importantes da Análise Comportamental que consiste em pequenos passos de aprendizado. Esses passos se encontram na plataforma GEIC. O jogo, porém, não possui nenhum aspecto de inteligência artificial que se adapte ao progresso do aluno. A Figura 10 mostra uma tarefa de aprendizado o qual se deve identificar a figura do tubo com a palavra “tubo”. Do lado há o quadro de atributos do jogador.

Figura 10 – Cena de uma tarefa de ensino do jogo ALE-RPG. Ao lado um quadro de atributos do jogador.



Fonte: Adaptado de (Siqueira, 2012).

2.4.3. Grapho Game

O desenvolvimento do trabalho de Lyytinen *et al.* (2009) teve como objetivo auxiliar crianças com problemas de dislexia a identificarem e assimilarem os sons das letras e palavras com a escrita que elas possuem. O estudo está sendo desenvolvido na Finlândia e estendido em países do continente africano como Zâmbia onde possuem forte apelo para o ensino da língua inglesa. O conjunto de jogos gratuitos, como visto na Figura 11, permite auxiliar a identificar letras básicas e a relação entre o som que cada letra possui progredindo posteriormente entre sílabas e palavras.

Figura 11 - Jogo Grapho Game desenvolvido para auxiliar o aprendizado de fonemas e escrita para crianças com dislexia e dificuldade no ensino.



Fonte: (Richardson & Latvala).

Com vários níveis e de forma gradual a criança é capaz de interpretar e construir um repertório de letras e palavras. São várias tarefas distribuídas em vários mini-jogos com a finalidade de prender a atenção da criança e não desmotivá-la com o processo de aprendizagem. Contudo isso não basta para manter a criança motivada. É necessário colocar retornos positivos a cada interação correta da criança e com elementos dinâmicos. O jogo ofereceu essa característica com algoritmos computacionais baseado na matemática. Porém, apesar do sucesso da pesquisa, o trabalho não propôs realizar, através de técnicas de inteligência computacional, a análise do comportamento da criança.

Figura 12 – Um mini-jogo onde há a emissão de fonema e a criança deve clicar na sílaba correspondente.



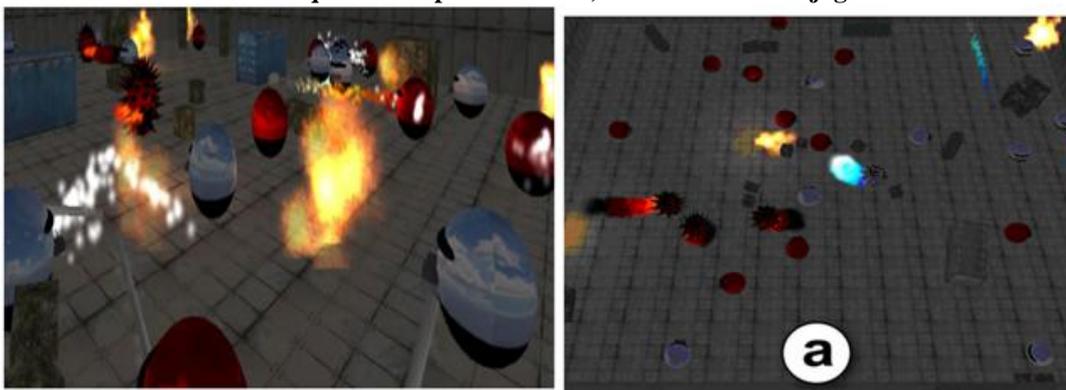
Fonte: (Richardson & Latvala).

2.4.4. Darwin Kombat

Outra perspectiva usada foi o estudo do jogo chamado *Darwin Kombat* criado por Machado *et al.* (2010). Esse jogo foi criado como experimento de teste de uma máquina de aprendizado baseado em classificação. Ele consiste em um jogo de estratégia onde os NPCs usam algum comportamento diante de uma emergência de ataque de um time inimigo. Para a criação do protótipo os desenvolvedores utilizaram várias tecnologias como o motor de jogo *Unity3D* aliado com o software WEKA. Como interface dessas duas tecnologias foi usado arquivos DLL juntamente com a biblioteca Java JNI.

O protótipo utilizou os algoritmos classificadores: C4.5, algoritmo tradicional baseado em árvores de decisões; *Naives Bayes*, que utiliza métodos probabilísticos eficientes e M5P, algoritmo para aprendizado de árvore de decisão que permite números reais para as classes estabelecidas. Este último, segundo o autor, foi o melhor algoritmo classificador.

Figura 13 - Jogo *Darwin Kombat* é um simulador de jogos de estratégia desenvolvido para experimentos com máquinas de aprendizado. a) Primeira zona do jogo.



Fonte: Adaptado de(Machado & Clua e Zadrozny, 2010).

Apesar de o jogo utilizar algoritmos classificadores o autor implementou técnicas de aprendizado por reforço com aprendizado supervisionado, isso porque ele utilizou o reforço como sendo o *score* (pontuação) de cada NPC. A cada rodada o classificador é repetidamente executado como estratégia de geração de novos e melhores agentes sem supervisão direta. Ou seja, o jogo realizou a simulação de comportamentos dos NPCs em momentos de emergências de ataque do time inimigo. Os NPCs, por sua vez, são classificados mediante certas ocasiões como: especialista em ataque curto (ninja) e especialista em ataque longo (tanque de guerra). Vale ressaltar que o jogo experimental não utilizou nenhuma característica da análise comportamental ou de cunho educacional, mas somente algoritmos de aprendizado já fixados pela literatura.

2.4.5. Estudo de personagens do tipo *Black and White*

No trabalho de Carvalho (2004) destaca-se no jogo comercial *Black and White* e seus NPCs com comportamentos inteligentes. O autor foca na inovação que o jogo proporcionou no começo do milênio. Nesse jogo, o jogador assume o papel de deus e cria civilizações com o objetivo de dominar as outras civilizações. O jogo tem o gênero de estratégia, mas um dos chamativos é a opção do jogador criar criaturas e treiná-las para agir de modos diversos a cada situação diferente. A criatura deve ser alimentada, enviada para as batalhas, recompensada com afago ou castigo a escolha do jogador.

O autor cita que o sucesso vem em parte por existir uma arquitetura deliberativa de agentes computacionais baseado no modelo de Crença, Desejo e Intenção. Neste trabalho o autor trata do problema de classificação (aprendizado supervisionado) utilizando instanciação baseado em regras de aprendizado geradas por árvores de indução no jogo.

Figura 14 - O Jogo comercial *Black and White* é considerado um jogo importante na área de vida artificial.



Fonte: Adaptado de (Carvalho, 2004).

2.4.6. Predição do Comportamento do Jogador em *Tomb Raider*

A pesquisa desenvolvida por Mahlmann *et al.* (2010) caracteriza-se pela exploração das possibilidades de previsão de aspectos do comportamento do jogador e do seu interesse no jogo *TombRaider: Underworld*. As previsões geradas são treinadas nos dois primeiros níveis do jogo. Uma das características da pesquisa foi identificar jogadores que, por sua vez, desistem de terminar os níveis propostos ou termina-os com sucesso. Logo o interesse do jogador no jogo é mapeado.

A pesquisa enfatiza o *Game Metrics Mining* ou métricas de mineração utilizadas em jogos onde é descrito algumas técnicas e métricas em jogos comerciais. Entre os jogos citados nessa pesquisa encontram-se: o jogo *Quake* que utilizou redes bayesianas; os jogos *War*

Galaxy e o *World of Warcraft* cujo utilizou as técnicas de *K-means* (clusterização) para suprir o aprendizado de NPCs unido com algoritmos de regressão e classificação.

No desenvolvimento do trabalho o autor começa a dar mais atenção às coleções de dados (*Data Collection*) onde ele subdivide em processamento de dados e extração das características. A pesquisa usou como base um ambiente chamado *SQE Suite* que captura e armazena métricas de games da empresa EIDOS. Esses dados podem ser armazenados e tratados por técnicas de IA e assim revelar características e perfis de jogadores em fases específicas. Os dados são diversos como: número de mortes, tempo de jogo, ajuda por demanda, tipo de morte (por queda, por inimigos ou pelo ambiente), capturas de tesouros, recompensas e configuração de mudanças (vida do inimigo, vida do jogador, quantidade de munição, número de *saves*).

Tabela 7 – Erro absoluto dos vários algoritmos de predição usado no trabalho baseado nas características do tempo para o 1º nível e 1º e 2º nível usando parâmetros ajustados. Menores valores são melhores.

Algorithm	Level 1	Levels 1 and 2
Simple linear regression	92.0	92.0
Multivariate linear regression	89.4	84.5
SMO Support vector machine	88.2	82.4
MLP/Backpropagation	107.2	111.5
REPTree decision tree (pruned)	92.5	91.8
Bagging REPTree (pruned)	85.2	83.5
M5Rules decision list	93.7	88.6
Gaussian processes	88.8	84.3
Baseline	100.0	100.0

Fonte: Adaptado de Mahlmann & Drachen *et al.*(2010).

Com o software WEKA (WEKA, 2012) foram testados os algoritmos de SMO (*Sequential Minimal Optimization*) e RETTree (algoritmo de árvore de decisão) que tiveram melhor performance de classificação, conforme a Tabela 7. Contudo o autor discute que os dados de treinamento foram pequenos para ter uma melhor avaliação. Novas variáveis e um maior número de dados poderiam dar uma realidade nova às informações do comportamento do usuário no jogo. O autor conclui que "*ideias sobre como manter os jogadores envolvidos são predominantes na indústria de jogos, e cada vez mais é apoiada pela psicologia comportamental e cognitiva visto que a pesquisa do usuário está a ganhar importância no desenvolvimento de jogos comerciais*" (Mahlmann & Drachen *et al.*, 2010).

2.4.7. Classificação de *gamebots* em um jogo digital de futebol com HMM

O estudo de Thureau *et al.* (2006) buscou fornecer um agente inteligente que aprende com o comportamento de humanos em um jogo eletrônico de futebol através da técnica Modelo Oculto de Markov (*Hidden Markov Model* - HMM) e métodos de Monte Carlo. Dessa forma

o autor destaca a possibilidade de classificar o comportamento de um NPC ou de uma equipe de NPCs via IA.

Além de classificar o comportamento, o *gamebot* (robô de jogo ou NPC) reconhece o comportamento do seu companheiro de time podendo gerar previsões e reconhecer padrões.

O autor destaca que: "*Jogos de computador tornaram-se um grande negócio com receitas anuais superiores aos da indústria do cinema mundial. No entanto, apesar do impacto previsível na qualidade de jogo e de vendas, personagens convincentes artificiais são raros*". O autor da pesquisa também aborda que o comportamento humano artificial em jogos é por muitas vezes declarativas pouco flexíveis ou com máquina e estados finitos sem criar a impressão de comportamento humano.

Na pesquisa foi utilizado um simulador de partidas de futebol do programa "*Tao of Soccer*" que possui dados de diversas partidas em sua base de dados. Seus experimentos classificaram 5 comportamentos entre dois *gamebots*:

- **driblando**: Um jogador "rouba" a bola e outro jogador fica seguindo-o conforme a Figura 15.
- **pegar-e-ir**: Um jogador pega a bola que é tocada imediatamente para outro jogador
- **recuperar bola**: Um jogador corre para a bola em repouso e imediatamente passa para o outro jogador.
- **passo longo**: A bola é tocada através de uma longa distância e não é retornada.
- **jogada solo**: O jogador com a bola não toca a bola, mas constantemente se move em uma direção, mas outro jogador permanece correndo ao seu lado.

Figura 15 – Pegar-e-ir no simulador de jogos de futebol *Tao of Soccer* como um comportamento classificado.



Fonte:(Thurau & Hettenhausen e Bauckhage, 2006).

2.4.8. Classificação de comportamentos via Rede Neural

No artigo de Puzenat & Verlut (2010) o autor demonstra que é possível classificar comportamentos de um indivíduo com aprendizado de máquina e uma rede neural devidamente ajustada. Sua abordagem foi, através de protótipos de jogos eletrônicos, mensurar ações do usuário coletando o que ele chama de diagnóstico. A rede neural utilizada foi o clássico MLP ou *Multi-Layer Perceptron* (*Perceptron* de Multi Camadas) com algoritmo de aprendizagem *backpropagation* programado através da biblioteca FANN (*Fast Artificial Neural Network*).

Figura 16 – Alguns minijogos utilizados nessa pesquisa.



Fonte: Adaptado de (Puzenat & Verlut, 2010).

O primeiro protótipo teve o objetivo de prever a idade mental do jogador utilizando diversos jogos educativos exibidos na Figura 16. Em seu experimento houve a simulação em tempo real e testes logo após alguns minutos de jogo. O melhor desempenho testado foi a previsão da idade do jogador depois de 10 minutos de jogo com uma precisão compatível com testes psicológicos.

O segundo protótipo utilizado buscou assegurar se o jogador é destro ou canhoto. Com um jogo de corrida exibido na Figura 17, a mesma rede neural do primeiro protótipo com pequenos ajustes e uma nova base de treino o experimento obteve 95% de sucesso.

Figura 17 – Foto do Jogo *open source Super TuxKart* utilizado nessa pesquisa.



Fonte: Adaptado de (Puzenat & Verlut, 2010).

2.4.9. Análise sobre os Trabalhos Relacionados

Os trabalhos anteriores e os pesquisados em literaturas da área possuem vários objetivos. Os trabalhos foram analisados conforme sua área de atuação e de forma sucinta. Este tópico visa analisar características comuns com o tema deste trabalho. Essa análise é efetuada em forma de tabelas buscando facilitar a interpretação de cada aspecto do trabalho.

A Tabela 8 apresenta uma agregação dos trabalhos pesquisados com os campos de atuação de interesse desta dissertação e onde cada trabalho relacionado se encaixa. O campo que possui um “X” representa a área que a pesquisa se manteve. Já a representação de um asterisco “*” representa que não houve interesse nesse campo de pesquisa, mas houve citações e partes pertinentes de interesse no trabalho.

Tabela 8 - Tabela de agregação dos trabalhos relacionados por campos de atuação

	Análise Comportamental	Jogos Digitais	Educação	Aprendizado de Máquina
GEIC	X			
ALE-RPG	X	X	X	
GraphoGame	X	X	X	
Darwin Kombat	*	X		X
Estudo de personagens do tipo Black and White	*	X		X
Predição do Comportamento do Jogador no jogo TombRaider	*	X		X
Classificação de gamebots em um jogo digital de futebol com HMM	*	X		X
Classificação de comportamentos via Rede Neural	X	X	X	X

A Tabela 9 mostra os eventuais pontos fortes e fracos, considerando os critérios de inovação nas áreas de: jogos digitais educativos e inteligência artificial para a educação, que são aspectos recorrentes dessa dissertação.

Destaca-se um asterisco (“*”) em alguns pontos fracos, contudo não foi encontrado um ponto fraco realmente significativo para estes trabalhos visto que eles não foram focados no ensino. Porém, pode-se considerar como um eventual “ponto fraco” a não utilização dessas pesquisas focadas em educação, pois esse é um assunto ligado a esta dissertação, aliado a análise comportamental e técnicas de inteligência computacional. Ou seja, esta dissertação procura contribuir cientificamente com estas características abordadas buscando remediar os pontos fracos e estabilizar alguns pontos fortes.

Tabela 9 - Tabela de comparação

Jogo	Ponto Forte	Ponto Fraco
GEIC	<ul style="list-style-type: none"> • Realiza o processo de aprendizagem e avaliação. • É estável e possui fácil agregação de novas tarefas de ensino no programa • Possui mecanismo de controle e análise de <i>feedback</i> através de consulta no banco de dados. • Utiliza aspectos da análise comportamental em seu processo de ensino. 	<ul style="list-style-type: none"> • Processo de aprendizado cansativo e tedioso. • Não possui nenhum aspecto de inteligência artificial. • Montagem das sessões de ensino consome tempo e esforço demasiado por parte dos psicólogos.
ALE-RPG	<ul style="list-style-type: none"> • Possui mecanismos de <i>feedback</i> apropriado. • Possuem Interface de fácil interação. • Passou por teste de usabilidade. • É considerado lúdico por agregar RPG com tarefas de ensino e reconhecimento de voz. • Executa passos de ensino exportados do GEIC. 	<ul style="list-style-type: none"> • Não possui nenhum aspecto de inteligência artificial. • O jogo de RPG em si é interrompido por tarefas de ensino obrigando o aluno a resolver as tarefas para avançar. Isso acarreta desinteresse no jogo com o tempo. • Não possui uma análise do aprendizado com inteligência computacional no jogo.
Grapho Game	<ul style="list-style-type: none"> • Foco na educação. • Auxilia crianças com problemas de dislexia ao redor do mundo. • Possui interface agradável e ilustrações atrativas para as crianças. • Utiliza aspectos da análise comportamental em seu processo de ensino. 	<ul style="list-style-type: none"> • Não possui nenhum aspecto de inteligência artificial. • Os mini-jogos não possuem conexão entre si havendo a necessidade de um tutor para explicar as tarefas ao aluno.
Darwin Kombat	<ul style="list-style-type: none"> • Possui uma arquitetura reaproveitável. 	<ul style="list-style-type: none"> • Não foi focado em educação*.

	<ul style="list-style-type: none"> • Os testes com aprendizado de máquina no contexto de agregar comportamento a grupos de NPCs foram de considerável importância. 	
Estudo de personagens do tipo Black and White	<ul style="list-style-type: none"> • Possui inovação na área de IA agregando comportamentos a NPCs 	<ul style="list-style-type: none"> • Não é focado em educação*.
Predição do Comportamento do Jogador no jogo <i>TombRaider</i>	<ul style="list-style-type: none"> • O trabalho utilizou diversos algoritmos de aprendizado de máquina para prever o comportamento de jogadores o que trouxe uma análise de desempenho desses algoritmos. • Utilizou uma base de dados real para fazer os seus testes que mostrou mais coerência em seus resultados. 	<ul style="list-style-type: none"> • Não é focado na educação*.
Classificação de <i>gamebots</i> em um jogo digital de futebol com HMM	<ul style="list-style-type: none"> • Classificou os <i>gamebots</i> com uma inteligência artificial baseada no Modelo Oculto de Markov e aprendizado de máquina. 	<ul style="list-style-type: none"> • Não é focado na educação*.
Classificação de comportamentos via Rede Neural	<ul style="list-style-type: none"> • Mostrou que um indivíduo pode ser classificado através do aprendizado de máquina. • Classificou a inteligência mental e se o indivíduo é destro ou canhoto para avaliar sua rede neural. 	<ul style="list-style-type: none"> • Os mini-jogos não possuem conexão entre si havendo a necessidade de um tutor para explicar as tarefas ao aluno. • Não houve foco na educação, mas sim na análise do algoritmo de rede neural.

3. Modelagem da complexidade de uma tarefa de ensino

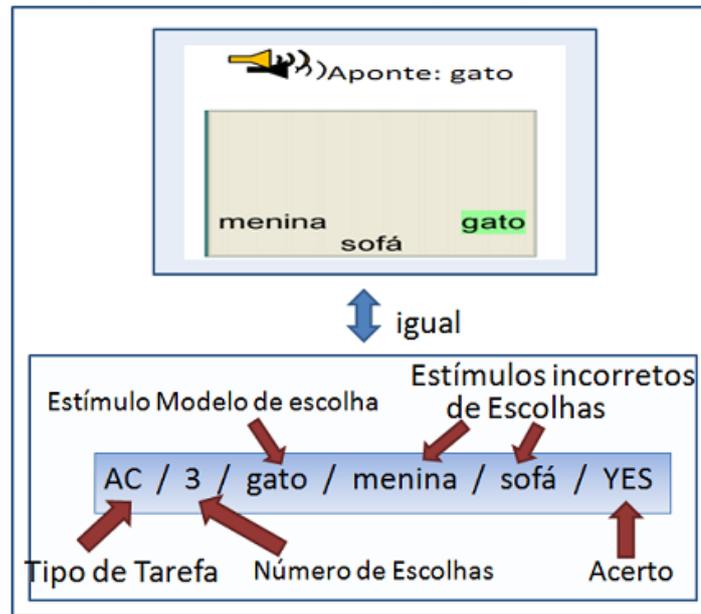
Este capítulo tem como objetivo explicar a complexidade de uma tarefa de ensino com seus devidos fatores de dificuldade. Apresentam-se as equações formuladas em conjunto com especialistas que representam a dificuldade das tarefas de escrita e leitura.

3.1. Dificuldade de uma tarefa de ensino

O programa de ensino citado anteriormente (ALEPP) possui diversos dados necessários que influenciarão na definição do aprendizado de máquina. Basicamente, para o sistema computacional proposto, uma tarefa de ensino é um conjunto de elementos, como mostra a Figura 18. Esses dados serão utilizados para analisar o comportamento do estudante. Eles serão coletados por sua vez e enviados para o modelo proposto a fim de iniciar o treinamento da IA e obter definições de padrões. Os principais dados são:

- **Acerto:** Variável booleana que indica acerto ou erro do aluno. Igual a 1 se o aluno acerta uma tarefa, caso contrário igual a 0.
- **Estímulo Modelo:** O estímulo modelo, ou simplesmente modelo, equivale à palavra que deve ser ensinada ao aluno na tarefa corrente. Toda tentativa ou tarefa tem um estímulo modelo que pode ser representado como figura, texto ou som.
- **Estímulos Escolhas:** Os estímulos escolhas são estímulos que serão incluídos em uma tarefa de ensino para testar a capacidade de identificação em relação ao estímulo modelo.
- **Número de escolhas:** É o número de estímulos de escolhas de uma tarefa. Neste trabalho utilizou-se de 1 a 5 estímulos de escolhas incluindo o estímulo correto. O estímulo correto é o estímulo que o aluno deve selecionar e está relacionado com o modelo de referência.
- **Tipo de Tentativa:** Esse campo representa a relação estabelecida entre os estímulos. Nele identificam-se os estímulos que são: som, figuras ou texto. Foram utilizados os do tipo **AB**, **AC**, **BC** e **CB** para identificar tarefas referentes de leitura, e **AE**, **BE** e **CE** para tarefas referentes de montagem das palavras ou escrita.

Figura 18–Esquema com os diversos dados que compõem uma tarefa de ensino.



Uma informação extraída dos dados apresentados acima servirá como parâmetro equivalente a **difficuldade de uma tarefa de ensino**, visto que, cada tarefa possui fatores de dificuldade que a tornam uma atividade mais complexa ou menos complexa. Objetivando identificar através de um valor numérico cada tarefa, no que se refere à dificuldade de ensino, foi formulado duas equações de dificuldades neste trabalho que serão descritas nos próximos sub-tópicos.

3.1.1. Dificuldade de Leitura

A fórmula da dificuldade proposta possui o intuito de definir um valor único de dificuldade por tarefa de ensino. Neste projeto considera-se que as tarefas que oferecem o aprendizado de leitura são as tarefas do tipo: AB, AC, BC e CB.

O cálculo da dificuldade da tarefa agrega os diferentes fatores de dificuldade inerentes de uma tarefa em um dado numérico. Esses fatores de dificuldade foram definidos juntamente com especialistas da área de psicologia educacional, de modo a obter um grau de complexidade da tarefa de ensino. Os fatores definidos foram:

- **Proximidade das palavras:** Para estabelecer uma métrica de dificuldade entre palavras semelhantes ou distintas, psicólogos especialistas na aquisição de leitura elaboraram uma tabela para classificar quanto uma palavra é semelhante à outra e com

base nestes campos, avaliaram as palavras propostas¹. A média das aproximações foi formatada na Tabela 10. O grau de proximidade entre as palavras varia no intervalo [0, 1]. Quanto maior o valor do grau de proximidade, maior a semelhança entre as palavras.

Tabela 10 – Proximidade das palavras. Amostra de 5 palavras².

	tomate	vovô	muleta	fita	pato
tomate	1	0,225	0,35	0,275	0,5
vovô	0,225	1	0,15	0,15	0,2
muleta	0,325	0,15	1	0,525	0,25
fita	0,275	0,15	0,525	1	0,375
pato	0,5	0,2	0,275	0,55	1

- **Tipo de tentativa:** para a relação inerente da dificuldade dos diferentes tipos de tarefas atribui-se um peso onde a soma desses pesos devem ser igual a 1 e cada peso deve ser maior que zero. Os pesos foram definidos conforme ilustrado na Tabela 11.

Tabela 11 – Pesos dos tipos de tentativas de leitura.

Tipo de tentativa:	AB	AC	BC	CB
Peso:	0,1	0,3	0,4	0,2

- **Número de escolhas:** Em uma tentativa BC, por exemplo, o modelo será uma figura e as escolhas serão palavras escritas em formato texto. Com isto, quanto maior o número de escolhas, maior a dificuldade de identificação entre a escolha correta e o modelo.

Baseado nos fatores de dificuldade citados acima foi elaborada a equação 1 que concentra essas variáveis em um único dado numérico.

$$Dr = \begin{cases} \frac{\sum_{i=1}^{n-1} Pin}{n-1} * \alpha + T * \beta + \frac{n}{n_{max}} * \gamma & , n > 1 \\ T * \beta + \frac{n}{n_{max}} * \gamma & , n = 1 \end{cases} \quad (1)$$

onde,

Dr: Dificuldade da tarefa de leitura,

Pin: fator de proximidade das palavras para escolhas incorretas em relação ao modelo,

T: peso do tipo de tarefa,

¹ No Apêndice A encontra-se o algoritmo de proximidade, que está em fase de testes. Ele foi formulado junto com psicólogos com base nas características de proximidades das palavras relevantes. No jogo projetado há a opção de utilizar a tabela de proximidade ou o algoritmo de proximidades de palavras.

² No Anexo A se encontram todas as palavras utilizadas nesta pesquisa.

n : número de escolhas disponíveis na tarefa,

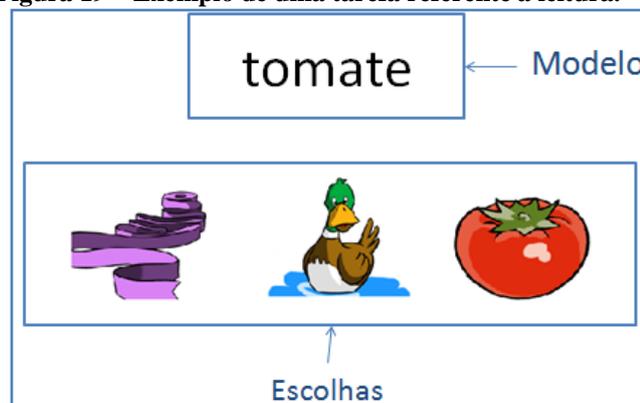
n_{max} : número máximo de estímulos de escolhas possíveis,

α , β e γ : peso dos fatores de dificuldade obtidos empiricamente.

Os valores α , β e γ são variáveis que podem ser ponderadas à medida que o aluno vai progredindo ou regredindo no seu aprendizado.

Para exemplificar a fórmula acima considere a tarefa da Figura 19 onde o modelo é a palavra em formato texto “tomate” e as escolhas de comparação são as imagens “fita”, “pato” e “tomate” da esquerda para direita.

Figura 19 – Exemplo de uma tarefa referente à leitura.



Agora considere que os pesos dos fatores de dificuldade α , β e γ foram sugeridos pelos psicólogos nos valores de: **0.25** (α), **0.25** (β), **0.5** (γ) respectivamente. A tarefa da Figura 19 é do tipo **CB** que tem peso **0.2** (T) e possui número de escolhas igual a **3** (n). Considera-se que o número máximo de escolhas possíveis para qualquer tarefa do sistema é igual a **5** (n_{max}). É necessário lembrar a Tabela 10 onde se encontram as proximidades das palavras. Nota-se que “**tomate**” possui valor de proximidade com “**fita**” igual a **0.275** e “**tomate**” possui valor de proximidade com “**pato**” igual a **0.5**. Com base nesses dados, a equação de dificuldade de leitura montada com esses valores é:

$$Dr = \frac{0,275 + 0,5}{3 - 1} * 0,25 + 0,2 * 0,25 + \frac{3}{5} * 0,5$$

O resultado de Dr , ou seja, a dificuldade dessa tarefa é **0.44688**.

3.1.2. Dificuldade de Escrita

A fórmula da dificuldade proposta para tarefas que vinculam-se a aquisição do aprendizado de escrita possui o intuito de definir um valor único de dificuldade por tarefa de ensino. Ela apresenta alguns termos da equação 1, mas possui termos novos visto que a aquisição da

montagem de palavras é diferente da aquisição de leitura. Considera-se que as tarefas que oferecem o aprendizado de escrita são as tarefas do tipo AE, BE e CE.

A diferença dos fatores de dificuldade desse tipo de aprendizado visa focar as escolhas de uma tarefa. Neste trabalho se concentrou esforços na montagem da palavra modelo através de sílabas exibidas para a interação do aluno. Logo as sílabas são as escolhas da tarefa. Esses fatores de dificuldade foram definidos juntamente com especialistas da área de psicologia educacional de modo a obter um grau de complexidade da tarefa de ensino. Os fatores definidos foram:

- **Escolhas incorretas em relação ao modelo:** Considera-se que uma sílaba é incorreta em relação ao modelo quando essa sílaba não está contida no conjunto de sílabas da palavra modelo. Por exemplo: a sílaba “la” não está contida no conjunto de sílabas da palavra “bo-lo”, logo ela é considerada uma sílaba incorreta. Diferentemente da sílaba “bo” da mesma palavra modelo, onde a primeira sílaba da palavra “bo-lo” é igual a sílaba em questão. Logo a sílaba “bo” é considerada uma sílaba correta. Portanto, na definição dos especialistas, quanto maior o número de sílabas incorretas, mais complexo se torna aquela tentativa.
- **Tipo de tentativa:** Para a relação inerente da dificuldade dos diferentes tipos de tentativas atribui-se um peso onde a soma desses pesos devem ser igual a 1 e cada peso deve ser maior que zero. Em uma tentativa BE, por exemplo, o modelo será uma figura e as escolhas serão sílabas escritas em formato texto. Logo os pesos foram definidos conforme a

Tabela 12.

Tabela 12 – Pesos dos tipos de tentativas.

Tipo de tentativa:	AE	BE	CE
Peso:	0,4	0,5	0,1

- **Número de escolhas da tentativa:** Para este fator de dificuldade, quanto maior o número de escolhas apresentadas ao aluno, maior a dificuldade de identificação e montagem da palavra entre a escolha correta e o modelo.
- **Número de escolhas do modelo:** Este fator de dificuldade refere-se a quantidade de escolhas disponíveis da palavra modelo. Quanto maior o número de sílabas a palavra possui, maior a dificuldade de identificação e montagem da palavra.

Baseado nos fatores de dificuldade citados acima foi elaborada a equação 2 que concentra essas variáveis em um único dado numérico.

$$Dw = \frac{S_i}{S_t} * \alpha + T * \beta + \frac{S_t}{S_{max}} * \gamma + \frac{S_m}{T_{sm}} * \sigma \quad (2)$$

onde,

Dw : Dificuldade da tarefa de Escrita,

S_i : Número de escolhas incorretas de sílabas em relação a palavra modelo,

S_t : número total de escolhas da tarefa,

T : peso do tipo de tarefa de Escrita,

S_{max} : número máximo de sílabas possível para uma tarefa,

S_m : número total de sílabas da palavra modelo da tarefa,

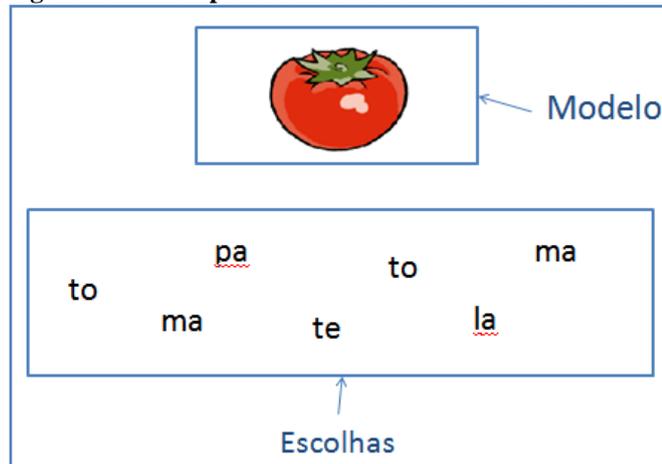
T_{sm} : número total de sílabas da palavra com maior número de sílabas do sistema,

α , β , γ e σ : peso dos fatores de dificuldade em relação às tarefas de escrita obtidos empiricamente.

Os valores α , β , γ e σ são variáveis que podem ser ponderadas à medida que o aluno progride no seu aprendizado.

Para exemplificar a equação acima considere a tarefa da Figura 20 onde o modelo é a palavra em formato de uma figura igual a “tomate” e as escolhas de comparação são as sílabas das palavras “tomate”, “mala” e “pato”.

Figura 20 – Exemplo de uma tarefa referente à escrita.



Considerando que os pesos dos fatores de dificuldade α , β , γ e σ foram sugeridos pelos psicólogos nos valores de: **0.3** (α), **0.2** (β), **0.2** (γ), **0.3** (σ) respectivamente. A tarefa da Figura 20 é do tipo **BE** que tem peso **0.5** (T). Observa-se que essa tarefa possui um total de **7** (S_t) sílabas para escolha do aluno. O número de sílabas da palavra modelo é **3** (S_m) e as sílabas “pa” e “la” são sílabas incorretas pois não são iguais a nenhuma das sílabas do modelo, totalizando **2** (S_i) sílabas incorretas. Suponha que a palavra com maior número de

sílabas do sistema (T_{sm}) é “muleta”, ou seja, **3** sílabas, e que o número máximo de sílabas que uma tentativa pode possuir seja **9**(S_{max}). Com base nesses dados a equação de dificuldade de escrita montada com esses valores é:

$$Dw = \frac{2}{7} * 0,3 + 0,5 * 0,2 + \frac{7}{9} * 0,2 + \frac{3}{3} * 0,3$$

O resultado de Dw , ou seja, a dificuldade dessa tarefa é **0.64127**.

Com esses dados consolidados é possível estimar o quanto uma tarefa é complexa para aquele aluno em valores numéricos baseados nos fatores de dificuldade provenientes do ALEPP.

4. Máquina de Aprendizado aplicada no Processo de Ensino

Este capítulo tem como objetivo demonstrar as principais características do projeto da máquina de aprendizado. Apresenta-se a técnica de aprendizado supervisionado utilizada e outras informações relevantes. Exibe-se também, com maior detalhamento, o programa de ensino mapeado pela máquina de aprendizado, que será utilizada pelos usuários do jogo.

4.1. Aprendizado Supervisionado para o Programa de Ensino

A análise comportamental é um dos campos da psicologia que está intrinsecamente ligada com o aprendizado de máquina no estudo da aprendizagem humana. Questões que envolvem a aquisição do “aprendizado” pelo computador e como os indivíduos aprendem possuem uma ligação interessante para pesquisas.

Apesar de existir certa dificuldade de compreender o aprendizado humano, o que dificulta os esforços de um tipo específico de técnica computacionalmente inteligente, estudos estão ganhando foco visto que a aprendizagem humana e a aprendizagem de máquina são próximas em questões científicas (Mitchell, 2006). Vislumbrando essa tendência, um dos objetivos deste trabalho é propor a integração de uma máquina de aprendizado com o método de ensino citado nos capítulos anteriores.

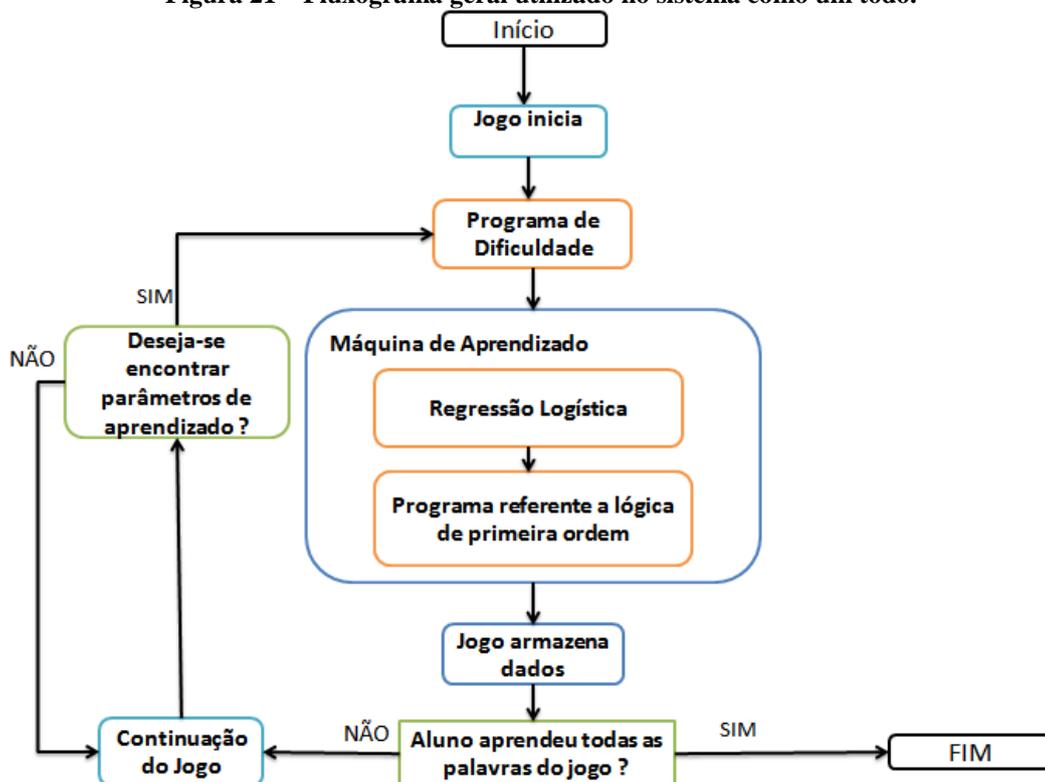
O cenário de aprendizado do ALEPP envolve alunos que possuem tarefas bem definidas a serem desenvolvidas e resultados bem consolidados para cada tarefa, além de possuir um supervisor humano que identifica os fatores de aprendizado de cada criança. Logo o tipo de problema de aprendizado de máquina que deverá ser abordado pode ser descrito como **aprendizado supervisionado**.

Diversos algoritmos são usados com a finalidade de solucionar o problema de aprendizado supervisionado como: rede neural (Faceli & Lorena *et al.*, 2011), *support vector machine* (SVM) (Faceli & Lorena *et al.*, 2011) e regressão logística (Gryczak & Lindebek *et al.*, 2010). Para este trabalho a **regressão logística** se mostrou mais adequada, pois engloba a particularidade de aprendizado supervisionado e possui vantagens interessantes a fim de ser executada tanto em dispositivos móveis como em computadores *desktops*. Em comparação com outras técnicas de classificação, as vantagens da regressão logística são: facilidade de programação, nível de complexidade relativamente baixo em tempo de execução e melhor

desempenho no número de acertos para problemas simples com poucos dados de treino, segundo Gryczak & Lindebck *et al.*,(2010).

A Figura 21 representa um fluxograma geral do sistema proposto para o cenário de ensino que será executado para a análise de comportamental de um participante apenas, ou seja, de forma individualizada. Cada estado está relacionado com um programa computacional desenvolvido no sistema. Pode-se verificar que o jogo inicia normalmente e logo em seguida há a instanciação e execução dos programas principais. O objetivo é encontrar parâmetros de aprendizados de cada palavra do jogo no que refere-se ao aprendizado de máquina. Logo, é necessário que o programa de dificuldade calcule valores de dificuldade para cada tarefa de ensino. Dentro da máquina de aprendizado a regressão logística é responsável por encontrar os coeficientes de aprendizado para previsão de acerto das tarefas de ensino. O programa referente a lógica de primeira ordem calcula dados estatísticos baseado nos dados gerados pela regressão logística e outros parâmetros auxiliares para classificação do aprendizado da palavra. O jogo se encarrega em armazenar os dados gerados e caso haja a necessidade, calcula-se novamente os parâmetros de aprendizado baseado em novos dados. O processo termina quando o aluno aprender todas as palavras do sistema.

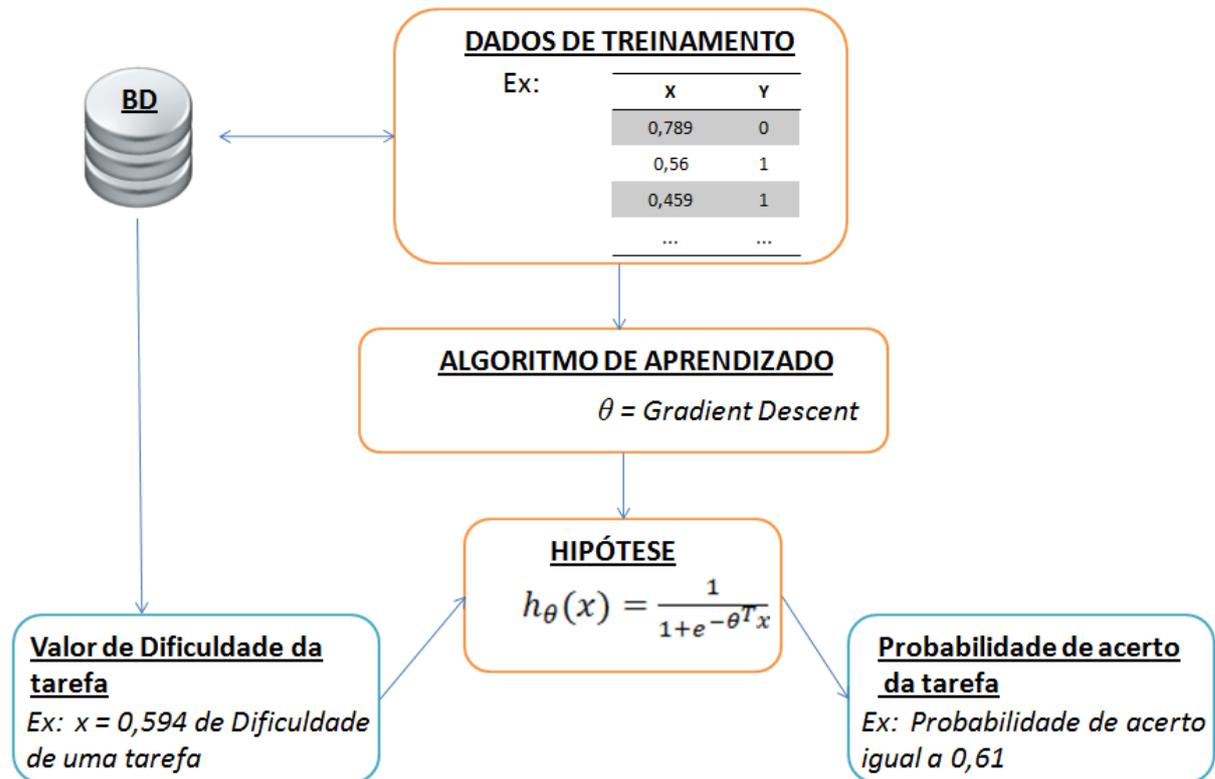
Figura 21 – Fluxograma geral utilizado no sistema como um todo.



O processo de aprendizagem de máquina para o problema proposto, referente a regressão logística, é realizado conforme a Figura 22, onde:

- Os **dados de treinamento** são utilizados como o histórico de ensino do aluno. Também conhecido como dados do Pré-teste, os dados de treinamento são necessários para “treinar” a máquina de aprendizado. Quanto maior a quantidade de dados, melhor será a precisão do sistema. Nesse caso, os dados necessários são os valores de dificuldade e o acerto ou erro da tarefa.
- O **algoritmo de aprendizado** é executado para definir os parâmetros necessários. Foi utilizado o algoritmo conhecido como *Gradient Descent* (Andrew, 2011), para encontrar os parâmetros (θ) baseados nos dados de treino. O algoritmo de aprendizado também utiliza a representação da hipótese, que é uma função $h(\cdot)$.
- Após a execução do algoritmo de aprendizado, o sistema gera uma **hipótese** com os parâmetros alinhados. A hipótese é representada pela função *sigmoidal* recorrente da regressão logística e explanada no próximo sub-tópico.
- Com a hipótese (função h) estabelecida, basta entrar com um **valor de dificuldade** qualquer, ou seja, o nível de complexidade de uma determinada tarefa.
- E a **probabilidade de acerto** é caracterizada como o resultado da hipótese para uma tarefa n . Em suma, é a consequência da probabilidade do aluno acertar dado um valor de dificuldade de uma determinada tentativa.

Figura 22 – Representação do processo de aprendizado de máquina no programa e ensino



Esse processo pode ser retreinado o que ajustará os parâmetros de aprendizado com novos valores caso novas informações sejam adicionadas na base de dados. Dessa forma, baseado nos dados de treinamento, é encontrada uma hipótese que pode ser utilizada para encontrar uma probabilidade de acerto ou erro de uma determinada tarefa com determinada dificuldade de ensino. Ou seja, pode-se destacar que os dados de treino são manipulados até a execução do algoritmo de aprendizado e logo após há a manipulação dos dados de teste com a hipótese da regressão logística, dados de entrada e saída. Nos próximos sub-tópicos serão descritos em detalhes as características da máquina de aprendizado juntamente com os conceitos pertinentes.

4.1.1. Representação da Hipótese da Regressão Logística

Nos modelos de regressão logística geralmente a variável de saída é uma variável binária (*true ou false*) e as variáveis de entrada podem ser, categóricas ou contínuas (Gryczak & Lindebck *et al.*, 2010). Casos de classificação de email como *spam* e *não spam* ou tumor maligno e tumor benigno são alguns exemplos em que as observações podem ser classificadas exclusivamente em 0 ou 1. A equação 3 mostra uma forma generalizada de classificação positiva e negativa.

$$y_{\text{aprendizado}} = \begin{cases} 0, \text{Classificação Negativa} \\ 1, \text{Classificação Positiva} \end{cases} \quad (3)$$

Assim como outros métodos de aprendizado de máquina, a regressão logística também possui uma **hipótese** (h) que é uma função que recebe como entrada um valor x e retorna um valor estimado y . Ou seja, a função h mapeia os valores de x para y . A hipótese da regressão logística é definida na equação 4,

$$h_{\theta}(\mathbf{X}) = g(\mathbf{X} \boldsymbol{\theta}^T) \quad (4)$$

onde:

h_{θ} : é a hipótese que representa a função de previsão e classificação dos dados.

\mathbf{X} : é a matriz com os dados de treino referentes à dificuldade das tarefas.

g : função sigmóide ou função logística.

$\boldsymbol{\theta}$: é um vetor com os coeficientes de aprendizado de máquina baseado nos dados de entrada.

Para este trabalho os dados de treino referentes à dificuldade de uma tarefa de ensino são representados através da matriz \mathbf{X} de ordem $m \times 2$ onde m é o número de exemplos de dados de treino. A primeira coluna desta matriz é igual a 1 e a segunda coluna são valores de dificuldade da tarefa, conforme ilustrado no exemplo da equação 5.

$$\mathbf{X} = \begin{bmatrix} 1 & 0.36625 \\ 1 & 0.6785714 \\ \vdots & \vdots \\ 1 & 0.5541667 \end{bmatrix} \quad (5)$$

A representação $\boldsymbol{\theta}^T$ é a matriz transposta de $\boldsymbol{\theta}$, como mostra o exemplo do esquema linear da equação 6.

$$\boldsymbol{\theta} = [\theta_1 \ \theta_2], \quad \boldsymbol{\theta}^T = \begin{bmatrix} \theta_1 \\ \theta_2 \end{bmatrix} \quad (6)$$

A execução da multiplicação dos dados de treino \mathbf{X} com os parâmetros $\boldsymbol{\theta}^T$, do exemplo exibido acima, resulta na equação 7 destacada a seguir.

$$\mathbf{X} \boldsymbol{\theta}^T = \begin{bmatrix} 1 & 0.36625 \\ 1 & 0.6785714 \\ \vdots & \vdots \\ 1 & 0.5541667 \end{bmatrix} * \begin{bmatrix} \theta_1 \\ \theta_2 \end{bmatrix} = \begin{bmatrix} \theta_1 + 0.36625 * \theta_2 \\ \theta_1 + .6785714 * \theta_2 \\ \vdots \\ \theta_1 + 0.5541667 * \theta_2 \end{bmatrix} \quad (7)$$

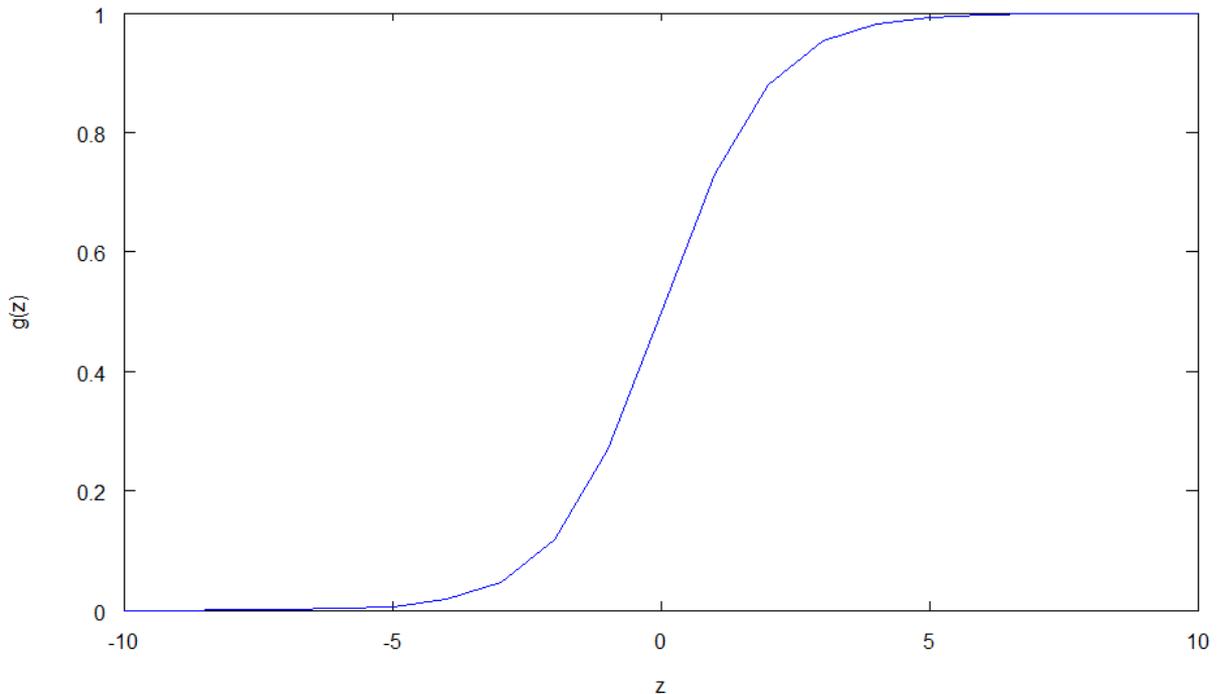
No modelo de regressão logística, também conhecido como modelo logístico, a função principal é baseado na função sigmóide. A função sigmóide g é representada na equação 8:

$$g(z) = \frac{1}{1+e^{-z}} \quad (8)$$

onde e é o número de Neper.

A função sigmóide varia entre 0 e 1, $0 \leq g(z) \leq 1$, o que torna o resultado uma probabilidade do evento ocorrer ou não ocorrer. A saída da função $g(\cdot)$ representa a probabilidade do evento ocorrer e z representa o valor que combina os fatores diversos do problema em questão. O Gráfico 1 ilustra uma função sigmóide utilizada nesse trabalho.

Gráfico 1 - Aspecto de uma curva sigmóide ou logística.



Baseado na hipótese da regressão logística e na função sigmóide descrita anteriormente, a interpretação nesse contexto define que $z = \mathbf{X} \boldsymbol{\theta}^T$. A hipótese apresenta uma nova configuração na equação 9.

$$h_{\boldsymbol{\theta}}(\mathbf{X}) = \frac{1}{1 + e^{\mathbf{X}(-\boldsymbol{\theta}^T)}} \quad (9)$$

Portanto, com base na representação exibida na equação 9, a meta do processo de aprendizado de máquina é encontrar e determinar o coeficiente $\boldsymbol{\theta}$ que é o parâmetro que determina a projeção da curva logística, possibilitando a predição do aprendizado de uma tarefa de ensino.

4.1.2. Treinamento da máquina de aprendizado

No tópico anterior foi explanada a representação da hipótese da regressão logística, contudo o parâmetro necessário para determinar a curva logística precisa ser definido a fim de prosseguir na predição dos dados e obter uma classificação razoável. Ou seja, é preciso encontrar um $\boldsymbol{\theta}$ aceitável baseado no histórico de resultados do aluno.

Neste trabalho, a matriz X possui os dados de treino correspondente às dificuldades das tarefas apresentadas aos alunos. Um dos atributos dessa matriz é a **dificuldade** gerada pelas equações de dificuldade de cada tarefa apresentadas nos tópicos anteriores.

Outro conjunto de dados de importância é o vetor y . Esse vetor armazena os dados que identificam a classe dos dados de treino, ou seja, o **acerto** ou **erro** de cada tarefa de ensino apresentadas aos alunos. O esquema abaixo da Tabela 13 exemplifica essa relação entre os dados de treino.

Tabela 13 – Exemplo de dados de treinamento essenciais de cada tarefa para o modelo de dados da regressão logística. Para o vetor y , 1 = acerto, 0 = erro.

Índice da tarefa(i)	Matriz X		Acerto e Erro da tarefa (Vetor y)
	Valores para normalização	Dificuldade da tarefa	
1	1	0.35625	1
2	1	0.6785714	0
3	1	0.38125	1
4	1	0.5541667	0
...
M	1	0.765286	1

Após a identificação do conjunto de dados é necessário executar o algoritmo de aprendizado para obter o parâmetro θ . Para este trabalho, o algoritmo de aprendizado utilizado foi o algoritmo de otimização denominado de *Gradient Descent*, que é um método utilizado para encontrar o mínimo local de uma função através de passos contínuos (Rauber, 2012). O algoritmo de *Gradient Descent* tem menor complexidade, o que facilita sua programação e execução em dispositivos móveis, sendo utilizada a versão proposta por Andrew (2011), conforme ilustra o pseudocódigo do algoritmo 1.

Algoritmo 1: Algoritmo <i>GradientDescent</i>	
1:	Repita de 1 até <i>NumIter</i> ,
2:	{
3:	$temp1 = \theta_1 - \alpha \frac{1}{m} \sum_{i=1}^m (h_{\theta}(\mathbf{X}^{(i)}) - \mathbf{y}^{(i)})X_1^{(i)}$
4:	$temp2 = \theta_2 - \alpha \frac{1}{m} \sum_{i=1}^m (h_{\theta}(\mathbf{X}^{(i)}) - \mathbf{y}^{(i)})X_2^{(i)}$
5:	$\theta_1 = temp1$
6:	$\theta_2 = temp2$
7:	}

Onde:

α : é a taxa de aprendizado do algoritmo,

m : é o número total de tarefas,

X : é a matriz de dados de ordem $m \times 2$ que possui a dificuldade de cada tarefa,

y : é o vetor com os acertos para cada tarefa,

i : é o índice das tarefas.

Para a execução do algoritmo de aprendizado, foi utilizada $NumIter = 500$. Tal valor foi selecionado após diversas execuções, onde foi constatado que um número de iterações grande implica uma demora na finalização da execução. Um número de iterações maior que 500 pode não melhorar os valores de saída.

A taxa de aprendizagem do algoritmo, α , foi selecionada empiricamente através da execução de diversos experimentos. O valor de α que se mostrou mais adequado foi igual a 8. Para valores menores a velocidade de convergência é reduzida, o que leva a uma demora na determinação de um valor satisfatório, porém com grandes chances deste valor ser preciso, ou seja, encontrar o mínimo local. Para valores grandes de α a convergência é prejudicada, pois é possível “saltar” a solução que indique um mínimo local.

Ao final desse algoritmo, determina-se os valores de θ baseado nos dados históricos de cada tarefa. Deste modo, obtém-se uma hipótese que nos faz lembrar a equação 9 resultando na probabilidade do aluno acertar ou errar uma tarefa baseado em uma dificuldade qualquer.

4.2. Modelo de máquina de aprendizado proposto

O processo de aprendizado de máquina explicado no tópico anterior é considerado o núcleo desse trabalho. Contudo, a classificação do aprendizado do aluno para as palavras não é determinada somente com a previsão de acerto e erro das tarefas. Há parâmetros auxiliares, que foram definidos em conjunto com os psicólogos e integrados ao modelo logístico de aprendizado de máquina, descrito com as demais características do programa de ensino.

4.2.1. Decisão do Aprendizado e parâmetros auxiliares

Existem algumas condições e variáveis extras para a decisão do aprendizado de uma palavra proposta, que são:

- **Condições Especiais (*CondEsp*):** as condições especiais são um conjunto de condicionais que retornam *true* ou *false* podendo ser adicionadas novas

condições conforme a necessidade do analista do comportamento. Essas condições especiais estão representadas na tabela a seguir.

Tabela 14 – Tabela com exemplos de condições especiais.

Condição	Descrição	Objetivo
$Q \geq 7$	Quantidade de tarefas realizadas (Q) maior ou igual a 7 por palavra.	Ter uma amostra mínima de repostas do aluno para executar o treinamento da <i>Machine Learning</i> .
$E \geq 1$.	Pelo menos uma tarefa do tipo Escrita (E) deve ser feita.	O aluno deve montar uma palavra com as sílabas, demonstrando habilidade de escrita.

- **Dificuldade Máxima (D_{max}):** é um valor máximo de dificuldade de uma tarefa para uma palavra modelo. Esse parâmetro é utilizado como limite de complexidade de uma tarefa de ensino para uma palavra específica. O sistema computa uma dificuldade máxima, através do programa de dificuldade, para cada palavra do sistema, tanto para questões de leitura quanto para escrita. Esse parâmetro possibilita encontrar a probabilidade de acerto para uma tarefa considerada a mais difícil do sistema.
- **Média da Dificuldade Geral (M_d):** é um parâmetro estatístico que busca encontrar a média da dificuldade das tarefas que obtiveram acerto. A condição para a média da dificuldade geral é $M_d \geq 50\%$ da D_{max} .
- **Taxa de Acerto (T_A):** equivale ao número de acertos dividido pelo total de tarefas executadas. Caso seja maior ou igual a 50% o seu valor é considerado *true*. A taxa de acerto também poderá ser utilizada como parâmetro de entrada para métodos vinculados com a geração de tarefas de ensino.
- **Probabilidade de acerto (*ProbA*):** probabilidade de acerto é gerada através dos métodos do programa da regressão logística conforme o algoritmo 2. Ela é definida pela hipótese determinada pela MA e por da dificuldade máxima. A probabilidade de acerto também poderá ser utilizada como parâmetro de entrada para métodos vinculados com a geração de tarefas de ensino.

Algoritmo 2 Algoritmo <i>Probabilidade de acerto</i>
1: ProbA = h_{θ} (50% da D_{max})
2: Se ProbA > 0,5 então,
3: {
4: True // Indivíduo Conhece Palavra
5: }
6: Senão,
7: {
8: False // Indivíduo Não Conhece Palavra
9: }

Com o objetivo de encontrar um valor booleano do **aprendizado da palavra**. Este trabalho propõe a utilização da equação 10, onde a probabilidade de acerto é computada através da regressão logística mostrada na equação 9 e algoritmo 2 para identificar se o aluno aprendeu ou não uma determinada palavra.

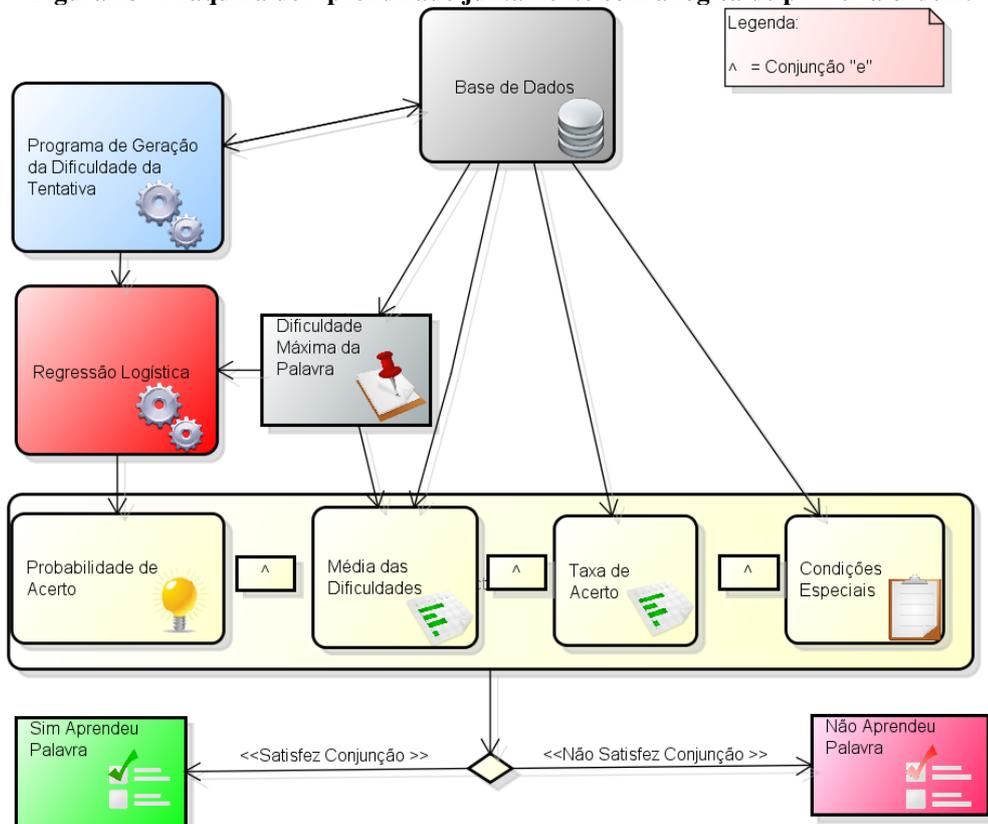
$$(ProbA > 0,5) \wedge (M_d \geq (D_{max} / 2)) \wedge (T_A \geq 0,5) \wedge CondEsp = True \quad (10)$$

Também é calculado o **grau de aprendizado da palavra** conforme a equação 11 e deste modo pode-se ter uma visão global de quanto o aluno aprendeu aquela palavra. Esse parâmetro pode ser calculado tanto para identificar a aquisição da leitura quanto para a aquisição da escrita.

$$GrauDeAprendizado = h_{\theta}(D_{max}) \quad (11)$$

Com esse grau de aprendizado da palavra e a lógica da primeira ordem, que determina se o aluno possui conhecimento suficiente da palavra, há a possibilidade de determinar dados mais próximos do real sobre o aprendizado do aluno na palavra treinada e testada. A Figura 23 ilustra a máquina de aprendizado integrada com a lógica de primeira ordem apresentada.

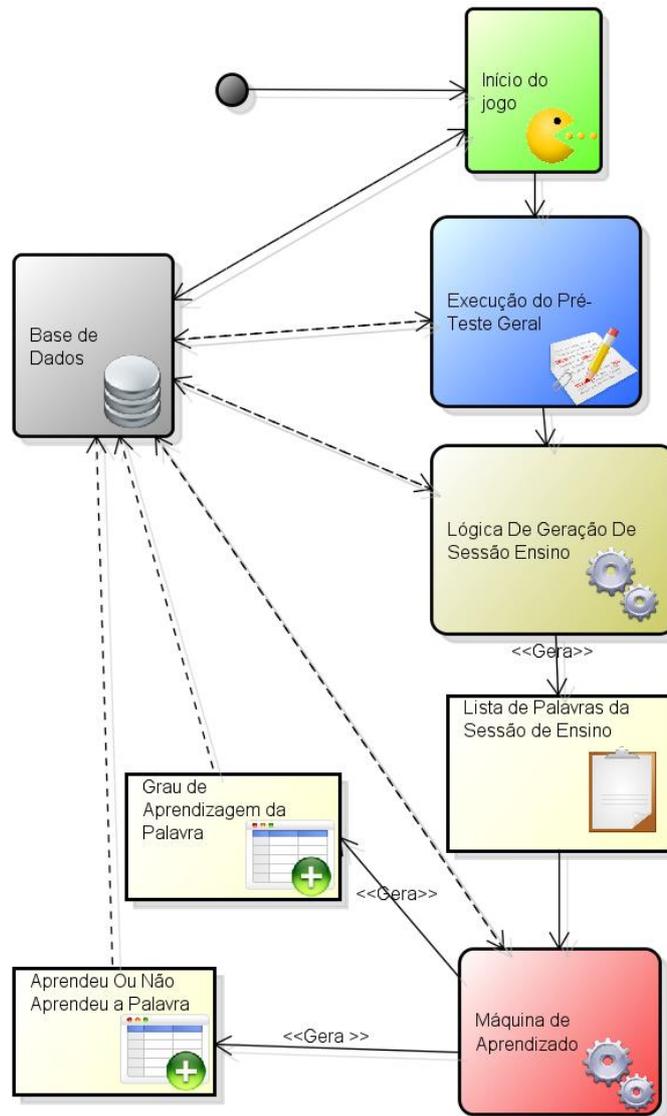
Figura 23 - Máquina de Aprendizado juntamente com a lógica de primeira ordem.



4.2.2. Protótipo do modelo de máquina de aprendizado

O protótipo da máquina de aprendizado desenvolvido para testes foi implementado na linguagem computacional Octave (GnuOctave, 2012) e mais adiante traduzido para linguagem C#. O protótipo da MA foi construído baseado na estrutura apresentada na Figura 24.

Figura 24 - Estrutura do protótipo da máquina de aprendizado.



Nessa estrutura nota-se que a base de dados possui transações de consulta, atualização e/ou inserção de dados representada pelas setas tracejadas. A atividade *Execução do Pré-Teste Geral* demonstra as primeiras apresentações de tarefas ao aluno, que devem estar armazenadas na base de dados para que o conjunto de respostas seja registrado.

A atividade *Lógica de Geração de Sessão de Ensino* é responsável por criar uma sessão de ensino para o aluno, ou seja, criar uma lista de palavras que o aluno ainda não

aprendeu. Esta atividade está em conexão com a *Base de Dados* para identificar qual palavra deve ser adicionada na *Lista de Palavras da Sessão de Ensino*.

A *Máquina de Aprendizado* tem como entrada a lista de palavras da sessão de ensino, onde cada palavra da lista passará pelo aprendizado de máquina a fim de enviar, para a base de dados, os resultados do grau de aprendizado da palavra e se o aluno aprendeu ou não a palavra. Dessa forma consolida-se o modelo de máquina de aprendizado proposto para o programa de ensino em questão.

5. Projeto de Jogo: As Aventuras de Amaru

Este capítulo tem o objetivo de mostrar as principais características do projeto do jogo em desenvolvimento, tais como: metodologia de desenvolvimento utilizada, enredo, ferramentas utilizadas, integração com a máquina de aprendizado, a geração de uma sessão de ensino dentre outras informações relevantes.

5.1. Visão Geral

O jogo desenvolvido nesta dissertação, denominado “As Aventuras de Amaru”, é um jogo estilo plataforma que constitui uma série de desafios encaixados à estrutura de ensino do ALEPP. Cada tipo de desafio é trabalhado separadamente. Esses desafios são chamados de mini-jogos com mecânica e solução própria. O jogador movimentará o personagem principal em um plano de duas dimensões, porém com gráficos compostos por objetos em três dimensões.

Nas primeiras reuniões com os psicólogos da UFSCar foi feita uma coleta de requisitos prioritários sobre o jogo. Muitos desses requisitos foram solucionados pela implementação do sistema inteligente visto no capítulo anterior e os outros por codificação do próprio jogo através de ferramentas que possibilitam diversos recursos. Embora esses requisitos fossem listados, os psicólogos deixaram a equipe de desenvolvimento livre para propor mecânicas, mini-jogos e inclusive a estória a ser montada pelo time, levando em consideração as características do público alvo. Na Tabela 15 estão listados os requisitos gerais deste projeto.

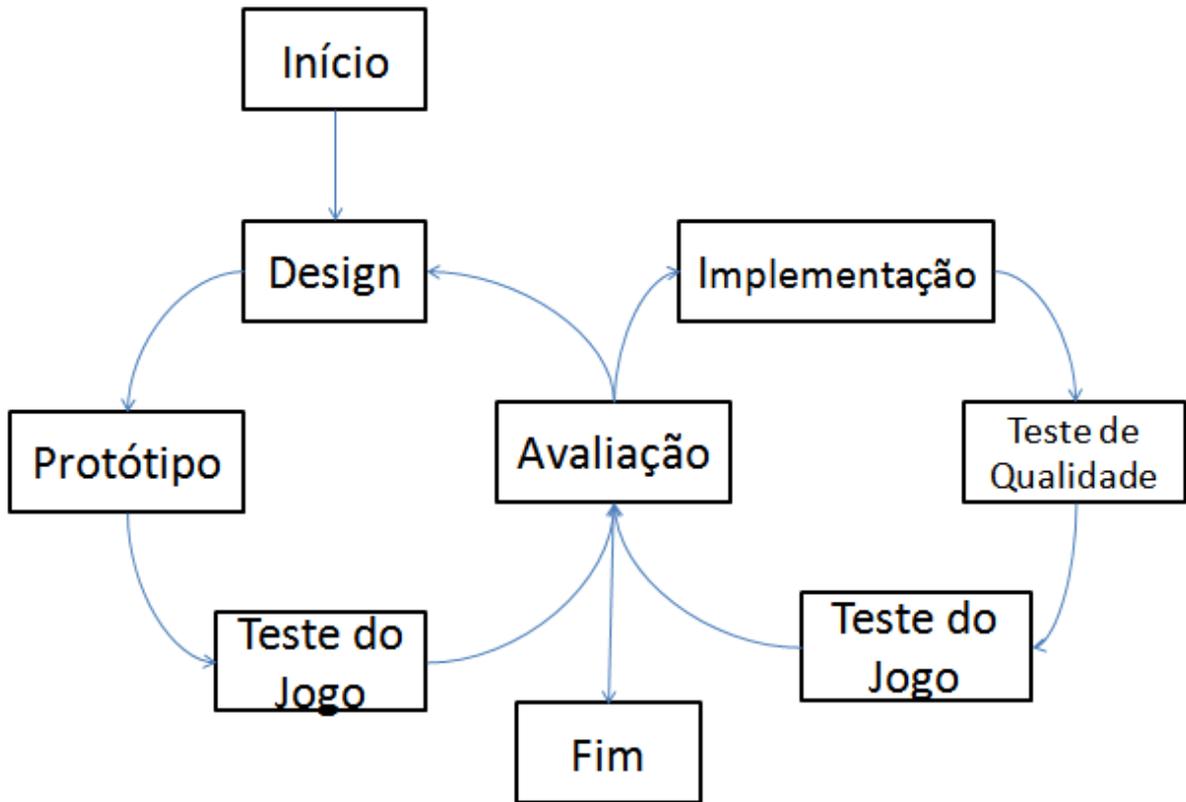
Tabela 15 – Requisitos gerais do Jogo

Nº	Descrição
R 01	Determinar quais palavras serão treinadas ou ensinadas por uma sessão de ensino.
R 02	Definir ordem das tarefas apresentadas para o aluno.
R 03	Palavras que estiverem no estado de “aprendidas” devem eventualmente ser apresentadas ao aluno.
R 04	A Estrutura da máquina de aprendizado deve seguir os conceitos da teoria da equivalência de estímulos e a teoria da programação do ensino vindouros da psicologia experimental cognitiva.
R 05	Avaliar qual o nível da criança.
R 06	Determinar um nível de dificuldade de cada tentativa.

R 07	Elaborar uma lógica que possibilite uma progressão da dificuldade das tarefas a fim de aumentar o nível de ensino da criança gradualmente.
R 08	Determinar uma ponderação/peso para os tipos de tarefas.
R 09	Utilizar uma estrutura similar com a do GEIC (Gerenciado de Ensino Individualizado por Computador).
R 10	Definir dificuldades para todas as palavras comparadas com as outras palavras.
R 11	Identificar quais palavras a criança já sabe ou não no decorrer do processo de aprendizado.
R 12	Os acertos ou erros da criança nos diversos tipos de tarefas devem ser armazenados para análise.
R 13	Ao final de cada sessão de ensino o desempenho da criança deve ser registrado para análise.
R 14	Agrupar os estímulos em tentativas ou tarefas.
R 15	Montar um padrão para ajudar a identificar quais os estímulos que são modelos e quais os que são comparações.
R 16	A seleção de comparações erradas serão tratadas como escolha errada, o que gera apenas a mudança para a tarefa seguinte.
R 17	O jogo deve ser controlado por mouse. Ou seja, o jogador deve interagir através do mouse em fases e desafios propostos na versão para PC.
R 18	Jogo deve ser desenvolvido primeiramente para computadores e em seguida para dispositivos móveis do tipo <i>tablets</i> com o sistema operacional <i>android</i> .
R 19	Visualização do desempenho dos alunos OFF-LINE

Em termos de metodologia de desenvolvimento, tanto para a máquina de aprendizado quanto para o jogo, foi utilizado a metodologia de prototipagem. Segundo Pressman (2006), a prototipagem tem a vantagem de ser utilizada quando o cliente não expõe de modo detalhado os objetivos do software ou o desenvolvedor está inseguro quanto a eficiência de algoritmos e operacionalidade do sistema como um todo. Algumas ponderações foram feitas sobre essa metodologia com o objetivo de se adaptar a realidade da equipe de desenvolvimento e das qualidades inerentes do desenvolvimento de jogos (Schreiber, 2011), como consta na Figura 25. Nela pode-se destacar que há um ciclo de projeto para um protótipo logo no início do processo determinando uma construção simples. Após há a avaliação e implementação seguido de testes contínuos para validar o protótipo. No centro temos a avaliação baseado nos testes de qualidade e da execução do jogo. Caso haja a necessidade de novas implementações, o ciclo é retomado para a construção de novos protótipos até a avaliação identificar que o jogo já está pronto, finalizando o processo.

Figura 25 – Interação com prototipagem rápida para jogos.



Fonte: Adaptado de (Schreiber, 2011).

5.2.Enredo e Personagens Principais

O jogo tem como personagem principal o explorador espacial chamado Amaru que, devido a problemas em sua nave, decide aterrissar no planeta Terra. Sua nave necessita de reparo, porém Amaru precisa arranjar peças de reparo o que o obriga a comunicar-se com os habitantes da Terra, ou seja, ele precisa saber ler e escrever no contexto do jogo. Para ajudá-lo existe o robô chamado Urama que o auxilia nessa aventura emitindo estímulos para seu aprendizado.

Figura 26 – Amaru e Urama.



No decorrer da sua jornada, Amaru irá aprender novas palavras e desta forma ganhar itens para consertar sua nave. O cenário inicial é uma fazenda que, no decorrer das fases irá sendo alterado entre floresta, indústria e cidade, como mostra os *backgrounds* da Figura 27.

Logo em seguida o leitor tem a visão do *gameplay* do jogo, conforme mostra a Figura 28 e Figura 29.

Figura 27 – Backgrounds dos cenários do jogo.

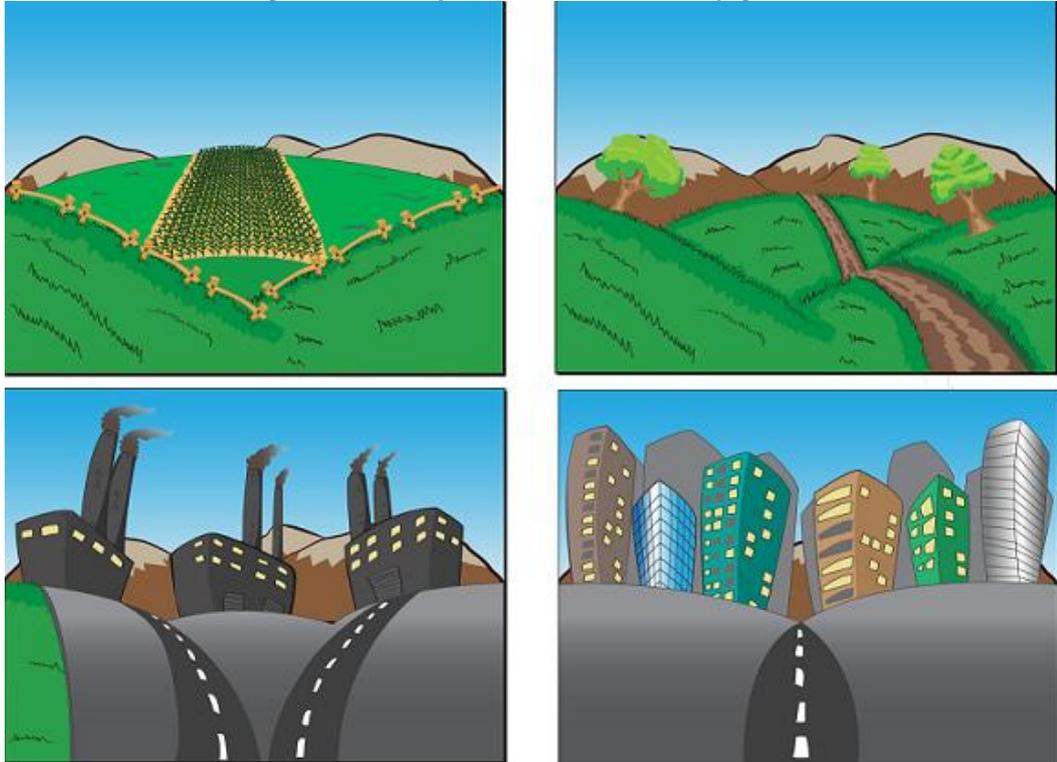


Figura 28 – Gameplay do jogo com tarefas referentes a leitura.

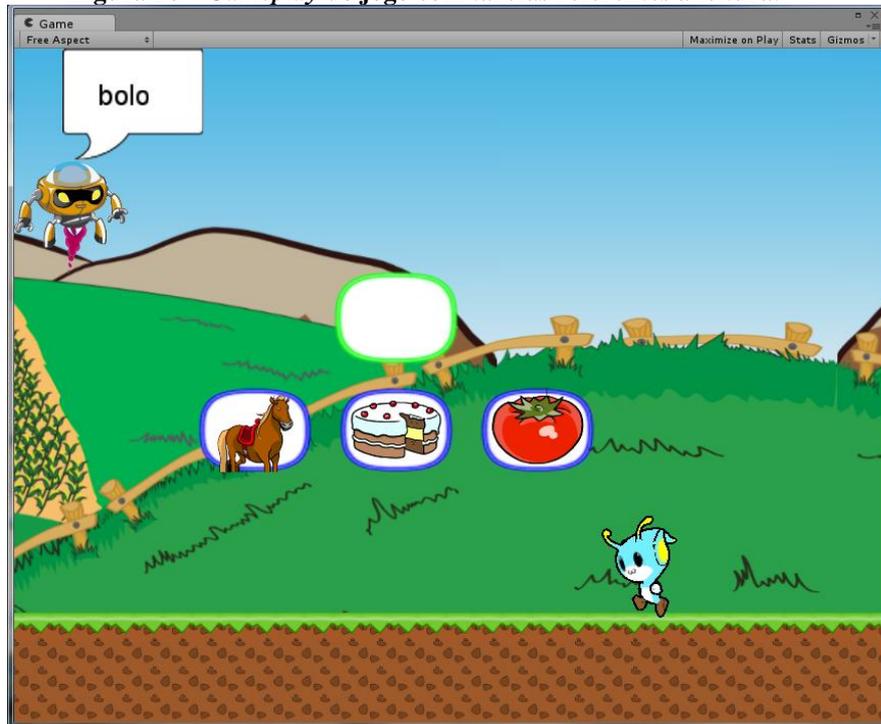
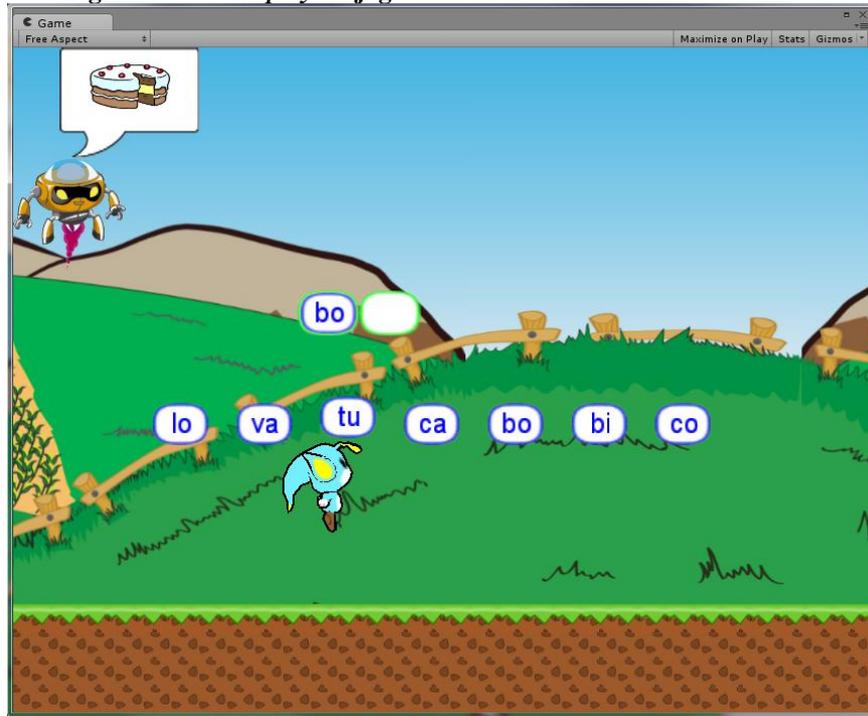


Figura 29 – Gameplay do jogo com tarefas referentes a escrita.



5.3.Ferramentas Tecnológicas

Essa pesquisa fez uso de diversas ferramentas para desenvolvimento do jogo. Juntas foram de fundamental importância no andamento desse projeto. As ferramentas utilizadas são:

- **Octave:** é uma linguagem interpretada de alto nível utilizada principalmente para cálculos numéricos (GnuOctave, 2012). Fornece recursos gráficos para manipulação de dados e sua sintaxe é bastante semelhante com o *Matlab* (MATLAB, 1994). Essa ferramenta foi utilizada no começo do projeto para obter uma visão mais detalhada da máquina de aprendizado e seu comportamento. Seu código foi traduzido logo em seguida para a linguagem de programação C# a fim de ser compatível com o compilador do *Unity3D*.
- **Unity3D:** foi a ferramenta principal para o desenvolvimento do jogo. O *Unity3D* é um motor de jogos proprietário com versões gratuitas e possui vários recursos que vão desde: suporte a *shaders* (técnica da computação gráfica usada para suavização de objetos e modelos 3D (Bell & Johnson, 2005)), programação em C# e *javascript*, compatibilidade de ferramentas de modelagem 3D, suporte a detecção de colisão e aspectos de física, entre outras características (Unity3D Support, 2012). Optou-se por essa ferramenta, além dos recursos já citados, pela capacidade de gerar executáveis do jogo em várias

plataformas como: *Playstation3*, *Nintendo Wii*, *Xbox360*, *Browsers*, Computadores, *tablets* com *Android* ou *iOS* (Sistema Operacional da Apple).

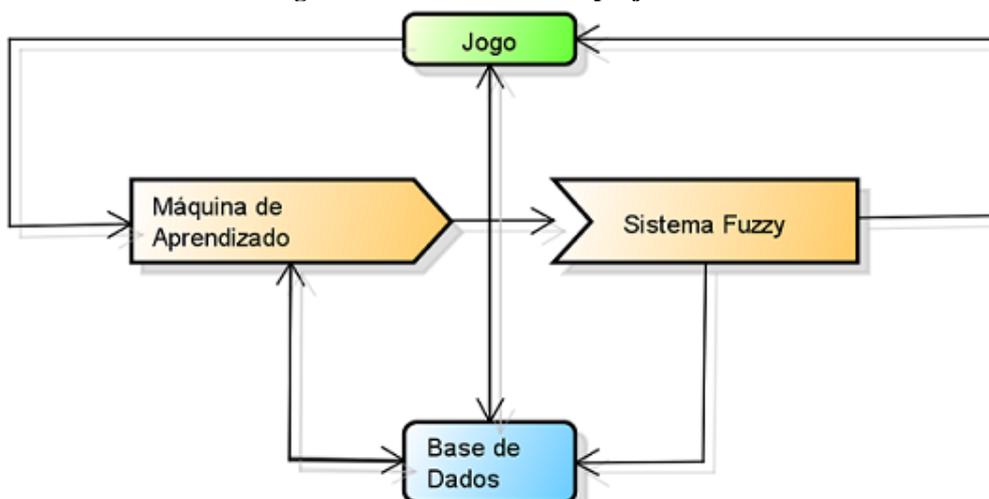
- ***MonoDevelop***: é um ambiente de desenvolvimento integrado com suporte as linguagens de programação como *C#*, *Boo* (uma variação da linguagem *Python*) e *javascript* (*MonoDevelop3.0*, 2012). Essa IDE é integrada ao *Unity3D* e por isso optou-se por sua utilização no que se refere a codificação do jogo.
- ***Microsoft Visual Studio .Net 2010 Express***: é uma IDE gratuita desenvolvida pela *Microsoft* para ser um ambiente leve da plataforma *.NET*. Sua principal função é fornecer à programação uma estrutura fácil de usar e de aprender (*Visual Studio*, 2010). Essa IDE foi usada para desenvolver o programa intitulado Relatório Amaru descrito no próximo tópico.
- ***Blender***: é uma ferramenta para modelagem e animação 3D (*Blender*, 2012). Essa ferramenta foi usada para modelar o personagem principal e fazer suas animações.
- **Editor de imagens *Inkscape***: é um software *opensource* para editoração de gráficos vetoriais e criação de imagens e documentos com esse formato. (*Inkscape*, 2012). Os gráficos de estímulos das tarefas e imagens do solo no jogo foram manipulados através desse programa.
- ***Astah***: é um software de modelagem UML com versões gratuitas (*Astah*, 2012). Foi utilizado para criar diagramas da UML e fluxogramas gerais para o melhor entendimento por parte da equipe de desenvolvimento e das características arquiteturais do jogo.

5.4.Lógica de Geração de Sessão de Ensino

O projeto do jogo envolve outra técnica de IA que se responsabiliza por gerar tarefas adaptadas ao aluno baseado em saídas da máquina de aprendizado. Uma visão global é apresentada na Figura 30 onde consta a integração da máquina de aprendizado com o sistema *fuzzy*³ (responsável pela geração de tarefas).

³ O sistema *fuzzy* faz parte de outro trabalho de dissertação desenvolvido por Adalberto Pereira Bosco, que agrega a característica de geração de tarefas adaptadas. Por essa razão o sistema *fuzzy* não será detalhado aqui.

Figura 30 – Visão macro do projeto.



A lógica de geração de sessão (Apêndice B) é um algoritmo que gera uma sessão de ensino para o aluno. Uma sessão de ensino equivale às palavras que devem ser ensinadas por meio de uma lógica estabelecida. Eventualmente uma palavra poderá ser apresentada novamente ao aluno para confirmar seu aprendizado, caso a máquina de aprendizado identifique que o aluno já possui conhecimento suficiente sobre a palavra. Essa fase é definida como Treino de Palavras ou Fase de Geração de Tarefas, onde o grau de aprendizado de cada palavra é fortemente levado em consideração.

Existem também algumas condições para transição da fase que estão intimamente ligadas a lógica de geração de ensino. Na Tabela 16, exibem-se as condições básicas referentes à mudança de fase e aprendizado em uma sessão de ensino.

Tabela 16 – Condições básicas para transição de fase

Descrição
Tempo de duração de uma sessão deve ter no máximo 40 minutos.
Mudar de mini-jogo a cada aprendizado de palavra.
Caso for constatado o aprendizado de 4 palavras em uma sessão, terminar essa sessão e continuar no próximo dia.
Caso for constatado o aprendizado de uma palavra, o jogador ganha uma peça de sua nave.
Se for constatado que o jogador não aprendeu nenhuma palavra na execução de 10 tarefas, mudar o mini-jogo.
A mudança de fase é feita com a constatação, por parte da máquina de aprendizado, que o aluno aprendeu 3 palavras por fase.

5.5. Programa: Relatório Amaru

O programa intitulado **Relatório Amaru** é um programa auxiliar do jogo que permite gerar um relatório com o desempenho do jogador. Ele foi construído pela equipe de

desenvolvimento e será utilizado por psicólogos como ferramenta auxiliar para acompanhar o aluno a fim de gerar informações e decidir a melhor estratégia de ensino para aquele aluno.

Basicamente o programa precisa do arquivo com os dados do pré-teste das tarefas e de todas as palavras do sistema como mostra a interface do programa na Figura 31. O programa então será capaz de gerar um relatório no formato de uma planilha para a verificação dos dados, como é exibido na Figura 32.

Figura 31 – Interface do programa Relatório Amaru.

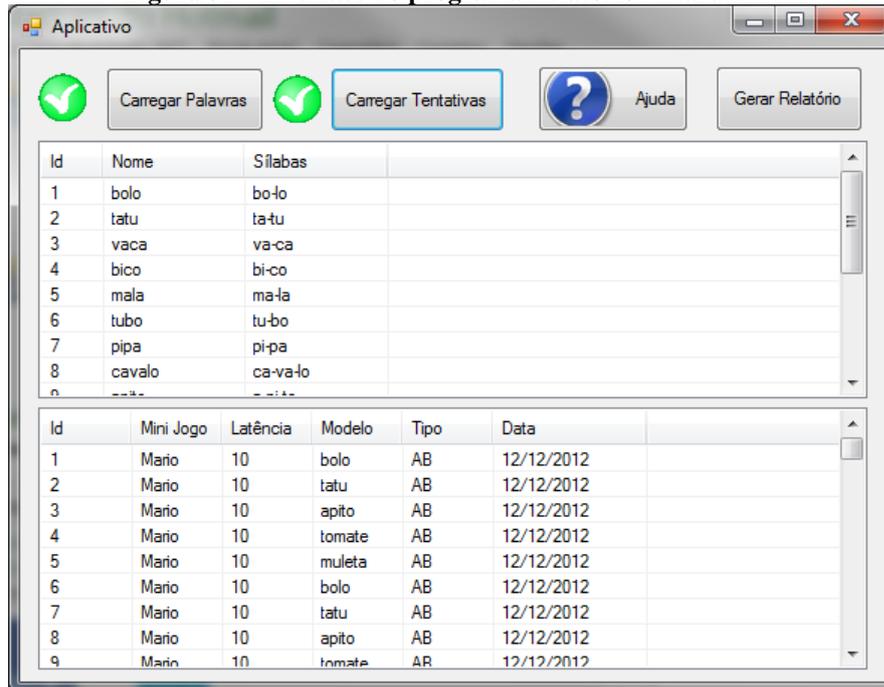


Figura 32 – Relatório gerado pelo programa.

	A	B	C	D	E	F	G	H	I	J	K
1	Participante:	Fulano									
2	Início:	12 12 2012									
3	Fim:	0 0 0									
4											
5											
6											
7	ETAPA1:	PRÉ - TESTE									
8											
9	Nro da Tent.	Mini-Game	Dificuldade da Tent.	Latencia.	Resultado.	Tipo de Tent.	Modelo.	Comp. 1	Comp. 2	Comp. 3	Comp. 4
10	1	Mario	0,35625	10	Acertou	AB	bolo	tubo			
11	2	Mario	0,325	10	Acertou	AB	tatu	pato			
12	3	Mario	0,2625	10	Acertou	AB	apito	luva			
13	4	Mario	0,428125	10	Acertou	AB	tomate	muleta	apito		
14	5	Mario	0,38125	10	Errou	AB	muleta	cavalo	apito		
15	6	Mario	0,3625	10	Acertou	AB	bolo	vaca	fita		
16	7	Mario	0,365625	10	Acertou	AB	tatu	luva	bolo		
17	8	Mario	0,5125	10	Acertou	AB	apito	tomate	cavalo	bolo	
18	9	Mario	0,5041667	10	Acertou	AB	tomate	fita	muleta	tatu	
19	10	Mario	0,5083333	10	Errou	AB	muleta	mala	tomate	apito	
20	11	Mario	0,3125	10	Acertou	AC	bolo	vaca			
21											
22											
23	Latência Máxima:	10	Latência Mínima:	0	Latência Média:	10					
24											
25	Nro Total de Tent:	70									
26	Tent. com Acertos:	21	Percentual Tent. com Acert	14,7 %							
27	Tent. com Erros:	49	Percentual Tent. com Erros	85,3 %							
28											
29	DADOS GERADO PELA IA DO PRÉ-TESTE										
30	ID da Palavra.	Palavra.	Silabas.	Dif. Máxima Escrita	Grau de Aprendiz. Escrita	Houve Aprendizado. Escrita	Dif. Máxima Leitura	Grau de Aprendiz. Leitura	Houve Aprendizado. Leitura		
31	1	bolo	bo-lo	0	0,1	0	0	0	0	0	0
32	2	tatu	ta-tu	0	0	0	0	0	0	0	0
33	3	vaca	va-ca	0	0	0	0	0	0	0	0
34	4	bico	bi-co	0	0	0	0	0	0	0	0
35	5	mala	ma-la	0	0,3	0	0	0	0	0	0
36											

Portanto, o projeto do jogo “As aventuras de Amaru” pretende tornar a estrutura de ensino mais amigável e interessante para o aluno acoplada a máquina de aprendizado que é executada de forma gerenciada e oculta para o jogador.

6. Experimentos e Resultados

Este capítulo descreve os experimentos e resultados deste trabalho. Os experimentos realizados foram divididos em três etapas, com o objetivo de validar esta pesquisa. Primeiramente foi feito uma validação das equações de dificuldade das tarefas de ensino através de um questionário enviado a especialistas. Numa segunda etapa foram realizados testes para validar o protótipo inicial da máquina de aprendizado com os especialistas da área de aprendizagem. Já na terceira etapa, com base nas análises realizadas na segunda e primeira etapa, realizou-se um experimento mais amplo, relacionando a leitura, escrita e a análise comportamental. A evolução dos parâmetros gerados estão ilustrados em gráficos para o cenário comportamental definido.

6.1. Validação das equações de dificuldade

O trabalho necessitou validar as equações de dificuldade das tarefas de aprendizado a fim de consolidar a viabilidade do projeto, explanados no Capítulo 3. Esta validação está relacionada com o programa gerador de dificuldade de uma tarefa em termos de protótipo do jogo. Este programa é responsável por gerar um valor de dificuldade para cada tarefa de ensino que é inserida no jogo, baseado nas equações de dificuldade.

A validação ocorreu com a submissão de questionários (Apêndice D) a um grupo de psicólogos contendo uma sequência de tarefas de ensino. Foi perguntado aos psicólogos qual seria o nível de dificuldade para cada tarefa de ensino ilustrado no questionário. O objetivo é identificar o nível de dificuldade para cada tarefa de ensino em uma escala definida (muito fácil = 1, fácil = 2, regular = 3, difícil = 4, muito difícil = 5).

A média do nível de dificuldade foi calculada levando em consideração as respostas dadas pelos psicólogos, como visto na Tabela 17. As linhas correspondem as tarefas apresentadas aos psicólogos e as colunas identificam as palavras ensinadas pelas tarefas de ensino onde os valores da tabela significam a diferença entre o nível de dificuldade da tarefa escolhida pelo psicólogo em relação ao nível de dificuldade da tarefa indicada pelo sistema formulado. Para a coluna interna que representa a diferença entre os resultados, o valor próximo de 0 sugere que a tarefa indicada pelo sistema formulado e pelos psicólogos tem o mesmo nível de dificuldade, enquanto que um valor próximo de 5 indica que as tarefas possuem níveis diferentes de dificuldade. Por exemplo, a tarefa 1 (linha 1) mostra que a palavra “bolo” (coluna 1) tem nível de dificuldade de valor 3, determinado pelo sistema. Psicólogos, através do levantamento realizado, avaliaram a tarefa com nível de dificuldade

igual a 1,5. Logo a diferença entre o valor gerado pelo sistema e o valor selecionado pelos psicólogos é igual a 1.5.

Tabela 17 – Nível de dificuldade das tarefas, dada a dificuldade gerada pelo sistema e a dificuldade sugerida pelo especialista. ESP: Especialista, SIS: Sistema Formulado, DIF: Diferença entre o sistema e a opinião do especialista.

	Bolo			tatu			apito			tomate			muleta		
	ESP	SIS	DIF	ESP	SIS	DIF	ESP	SIS	DIF	ESP	SIS	DIF	ESP	SIS	DIF
Tarefa 1	1,5	3	1,5	1,5	3	1,5	1,5	1	0,5	1,5	4	2,5	1,5	3	1,5
Tarefa 2	2,5	3	0,5	1,5	3	1,5	2	4	2	2	3	1	2	4	2
Tarefa 3	2,5	3	0,5	2,5	3	0,5	2,5	3	0,5	2,5	3	0,5	2,5	3	0,5
Tarefa 4	3	4	1	3,5	3	0,5	3,5	4	0,5	3	4	1	3,5	4	0,5
Tarefa 5	4	5	1	4	4	0	4	4	0	4	5	1	4	4	0
Tarefa 6	3	4	1	3,5	4	0,5	2,5	3	0,5	2,5	4	1,5	2,5	3	0,5
Tarefa 7	3	4	1	3,5	3	0,5	3,5	4	0,5	3	3	0	3,5	4	0,5
Tarefa 8	1	5	4	1	5	4	2,5	5	2,5	1,5	5	3,5	1,5	5	3,5
Tarefa 9	4	5	1	4,5	5	0,5	4,5	5	0,5	4	5	1	4	5	1
Tarefa 10	4	5	1	4	5	1	4,5	5	0,5	4,5	5	0,5	4,5	5	0,5
Tarefa 11	5	5	0	5	5	0	5	5	0	4	5	1	4	5	1
Tarefa 12	3	5	2	4,5	5	0,5	4,5	5	0,5	4	5	1	4,5	5	0,5
Tarefa 13	5	5	0	4,5	5	0,5	3,5	5	1,5	4,5	5	0,5	4,5	5	0,5
Tarefa 14	4	5	1	4,5	5	0,5	4,5	5	0,5	5	5	0	5	5	0

De acordo com a Tabela 17, pode-se analisar que:

- As tarefas consideradas complexas pelo sistema foram também consideradas complexas pelos psicólogos.
- Em todas as palavras da tarefa 8, a diferença no nível de dificuldade foi equivalente a 4. Isso porque essas tarefas são tarefas do tipo “Cópia”, onde o modelo é texto e as escolhas são sílabas para montagem do modelo (Tipo CE). Como melhoria para esse tipo de tarefa foi aconselhado um novo ajuste no peso desse tipo a fim de ajustá-lo a opinião dos especialistas.
- 14.28% das tarefas foram classificadas como equivalentes tanto pela IA quanto pelos psicólogos com mesmo nível de dificuldade. Isso significa que o nível de dificuldade selecionado pelos psicólogos foi o mesmo nível de dificuldade gerado pelo sistema.

- 64.28% das tarefas foram classificadas com uma diferença entre 0 e 1 ponto, o que significa que uma pequena diferença é aceitável entre a escolha feita pelo sistema e a escolha feita pelos psicólogos.
- 78.56% das tarefas apresentadas para os psicólogos, em relação ao seu nível de dificuldade, são similares e aceitáveis vistas a classificação de dificuldade gerada pelo sistema formulado.

6.2. Experimento da análise do aprendizado da leitura

Os primeiros testes foram efetuados com base na análise de relatórios de desempenho no ALEPP executados por meio do GEIC. Esses relatórios representam as sessões de ensinamentos já realizadas por crianças no programa de ensino (Anexo B). Com esses dados buscou-se avaliar o protótipo da máquina de aprendizado além de gerar alguns resultados preliminares.

Os primeiros testes foram aplicados com 4 especialistas do Laboratório de Estudos do Comportamento Humano da UFSCar. Vale ressaltar que foram levadas em consideração somente informações referentes às tarefas de leitura avaliando esse aprendizado. A palavra indicada no experimento foi a palavra “**muleta**”. A tarefa de dificuldade máxima (D_{max}) que ensina esta palavra é igual a **0,73594**. Foram induzidos 8 erros e 10 acertos nas tentativas. Duas simulações foram realizadas com os mesmos dados de treino para ambos, com seqüências de erros e acertos diferentes. Os resultados das dificuldades das tarefas são mostrados no Gráfico 2 e no Gráfico 3.

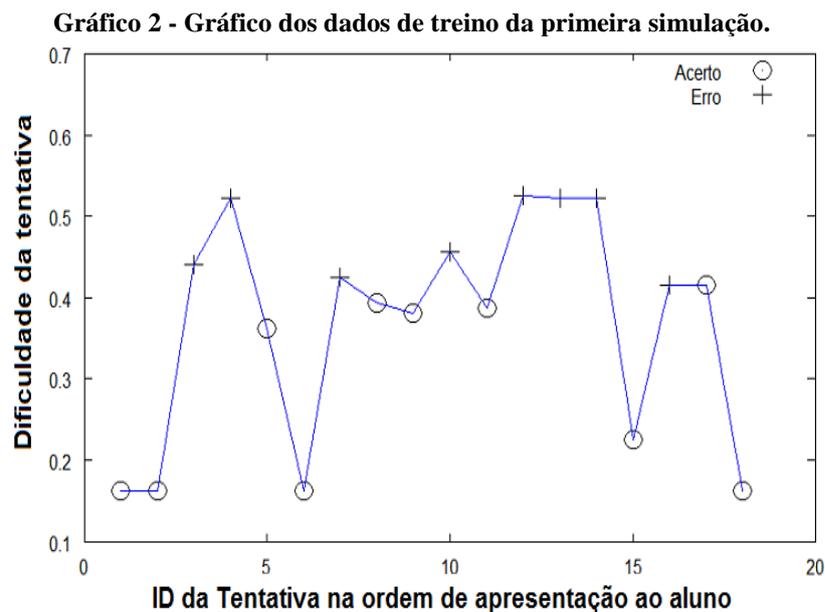
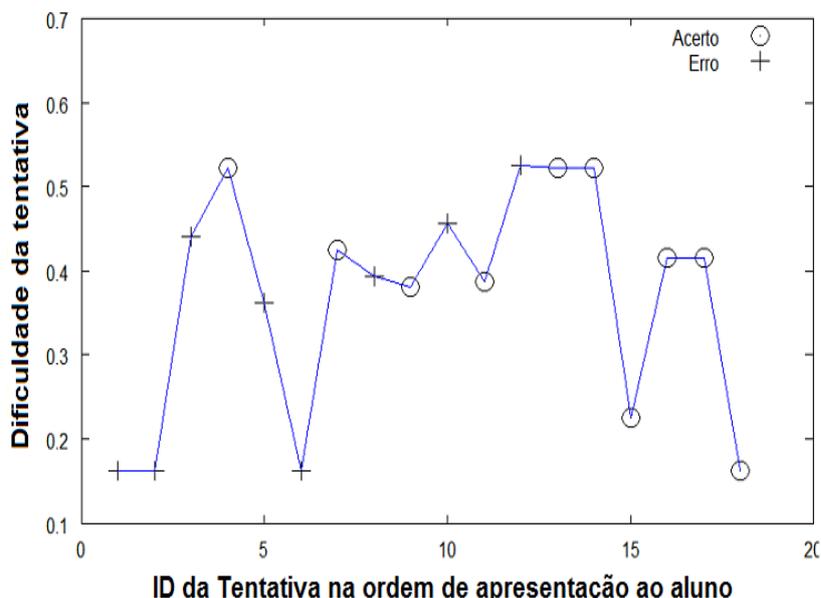


Gráfico 3 - Gráfico dos dados de treino da segunda simulação.



A primeira simulação consiste em induzir erros para as tentativas que contêm maiores dificuldades. Desta forma busca-se simular possível déficit de aprendizado de um aluno que tem dificuldade de acertar tarefas complexas. Para os dados do Gráfico 2 o programa gerou alguns dados, como consta na Tabela 18, onde pode-se observar que o indivíduo não aprendeu a palavra e o grau de aprendizagem adquirido na palavra é muito pequeno. Isso indica provável déficit de aprendizado. Classifica-se a palavra *muleta* como palavra que o aluno ainda não aprendeu.

O segundo experimento consiste em aplicar alguns erros iniciais e à medida que as tentativas são expostas, diminui-se o número de erros gradativamente até que por fim, no final da sessão, obtenham-se somente acertos independentemente da dificuldade da tentativa. Dessa forma busca-se simular o aprendizado gradativo do aluno. Para os dados do Gráfico 3 o programa gerou algumas informações, onde a palavra **muleta** é classificada como palavra que o aluno aprendeu e obteve grau de aprendizagem de cerca de 83%, conforme a Tabela 18 e relembrando os parâmetros auxiliares⁴ da lógica de primeira ordem do capítulo 4.

⁴ Para esse experimento foi omitido os resultados do modulo referente às Condições Especiais (*CondEsp*) citado na seção 4.2.1, visando uma melhor praticidade na visualização dos dados.

Tabela 18 – Tabela dos resultados da primeira e da segunda simulação

Simulação	Se ($M_d \geq 50\%$ da D_{max}) True Senão False	Se ($T_A \geq 0,5$) True Senão False	Se ($ProbA > 0,5$) True Senão False	Aprendeu Palavra (AP)	Grau do Aprendizado da Palavra (GA)
1ª	$M_d = 0,28156$ Então, False.	$T_A = 0,55556$ Então, True.	$ProbA = 0,99880$ Então, True.	False	3,8005E-21
2ª	$M_d = 0,39781$ Então, True.	$T_A = 0,55556$ Então, True.	$ProbA = 0,55653$ Então, True.	True	0,83954

Para validação do trabalho utilizou-se o questionário encontrado no Apêndice C expondo o Gráfico 2 e o Gráfico 3 a especialistas da área da análise comportamental da UFSCar a fim de classificarem a evolução do aprendizado da palavra. Segundo a escala de concordância adotada (insuficiente=1, regular=2, bom=3, ótimo=4, excelente=5) foi realizada uma distribuição em percentual correspondente o que resultou nas informações da Tabela 19, onde GA significa Grau de Aprendizagem e AP se houve Aprendizado na Palavra. Nota-se que os dados dos especialistas na área de análise comportamental estão de acordo com os dados de saída da máquina de aprendizagem, validando seus valores de saída.

Tabela 19 – Tabela dos resultados classificados pelos psicólogos para as simulações

	1º Especialista		2º Especialista		3º Especialista	
1ª Simulação	GA = 40%	AP = NÃO	GA = 40%	AP = NÃO	GA = 40%	AP = NÃO
2ª Simulação	GA = 60%	AP = SIM	GA = 80%	AP = SIM	GA = 80%	AP = SIM

6.3. Experimento simulado sobre a análise comportamental de alunos

Com base na aceitação dos testes da primeira e segunda etapa, foi dado continuidade para ampliar o trabalho com novos testes, agora levando em consideração as tarefas de escrita e leitura além de integrar a saída do aprendizado de máquina com um sistema *fuzzy* que se responsabiliza em gerar tarefas adaptadas ao aluno (Pereira, 2013).

6.3.1. Metodologia do experimento dos comportamentos de alunos

Com o objetivo de avaliar a máquina de aprendizado levando em consideração os primeiros testes e aliado ao jogo desenvolvido, foi primeiramente realizado simulações por parte dos psicólogos dos seguintes grupos de comportamentos:

- Alunos com Déficit de Aprendizado (DAP).
- Alunos com Aprendizado Gradativo (APG).
- Alunos com Aprendizado Consolidado (APC).

Os dados de cada simulação estão no Anexo C para consulta, onde pode-se analisar que os psicólogos induziram erros e acertos para um mesmo conjunto de tarefas determinando as simulações dos comportamentos propostos. O experimento utilizou 15 (quinze) palavras do Pré-Teste da Unidade 1 do programa GEIC. As 15 (quinze) palavras são: **bolo, tatu, vaca, bico, mala, tubo, pipa, cavalo, apito, luva, tomate, vovô, muleta, fita e pato.**

Das 15 palavras que correspondem a sessão de ensino, foi analisado o aprendizado somente de 5 (cinco) palavras devido a praticidade de trabalhar com amostras menores de dados. São elas: **bolo, tatu, apito, tomate e muleta.** O objetivo é classificar para cada uma das cinco palavras, um valor de aprendizado nas simulações propostas.

Outro objetivo importante é gerar, através da máquina de aprendizado e do sistema *fuzzy*, uma tarefa adequada para o aluno sendo ele um aluno com déficit de aprendizado, aprendizado gradativo ou aprendizado consolidado. Ou seja, se o indivíduo possui déficit de aprendizado, o sistema inteligente (máquina de aprendizado + sistema *fuzzy*) deve propor uma tarefa com nível de complexidade reduzido. Em outro caso, por exemplo, se o aluno permanecer com um comportamento gradativo de aprendizado, o sistema inteligente pode gerar uma tarefa com um nível de complexidade um pouco maior a fim de o instigar a continuar o aprendizado com desafios maiores.

As tarefas do sistema foram validadas através do questionário encontrado no Apêndice D e enviado para um grupo de psicólogos. Foi exibido uma sequência de tentativas para as simulações tarefas propostas. Os psicólogos preencheram o questionário indicando entre cinco opções uma tarefa adequada para o aluno baseado no desempenho realizado nas tentativas apresentadas.

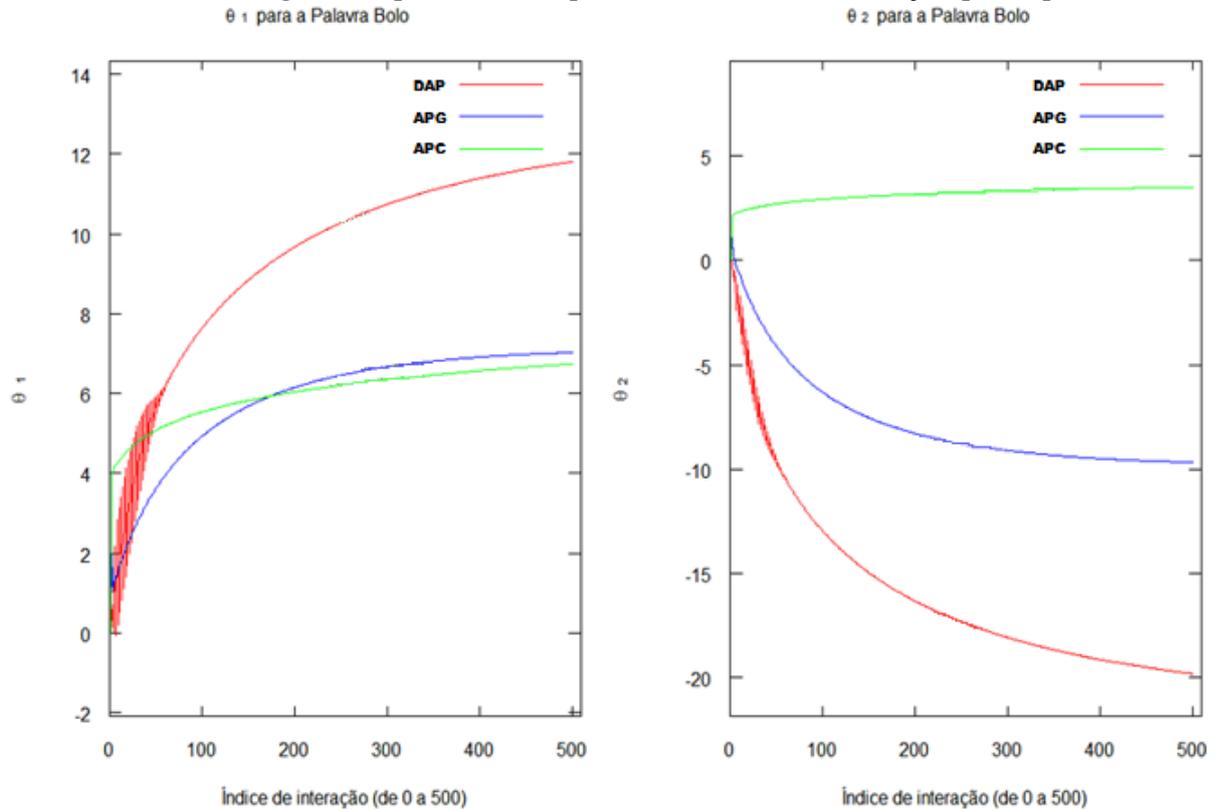
6.3.2. Resultado e Análise dos parâmetros da Máquina de Aprendizado

As simulações realizadas pelos especialistas são equivalentes ao pré-teste da primeira unidade do programa de ensino. Para a máquina de aprendizado esses dados são considerados os dados de treinamento. O algoritmo de aprendizado *Gradient Descent* irá gerar o parâmetro de aprendizado do algoritmo (θ) baseado nos dados de treino.

O parâmetro θ possui dois valores: θ_1 e θ_2 , explanados na seção 4.1.1. Observa-se a convergência e evolução desse parâmetro de aprendizado para os três comportamentos simulados de cada palavra do experimento, ilustrados no Gráfico 4 ao Gráfico 8.

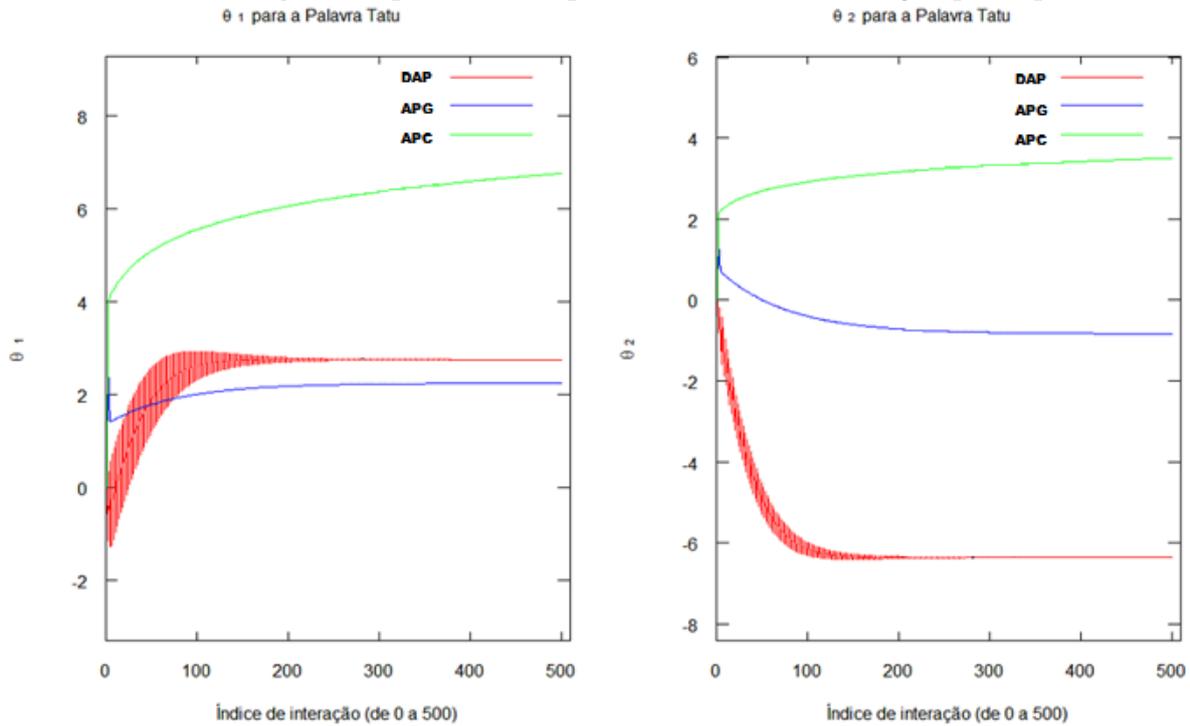
No Gráfico 4 pode-se observar que há uma convergência dos valores tanto de θ_1 e θ_2 para os comportamentos da simulação para 500 interações.

Gráfico 4 – Convergência do parâmetro de aprendizado θ das três simulações para a palavra Bolo.



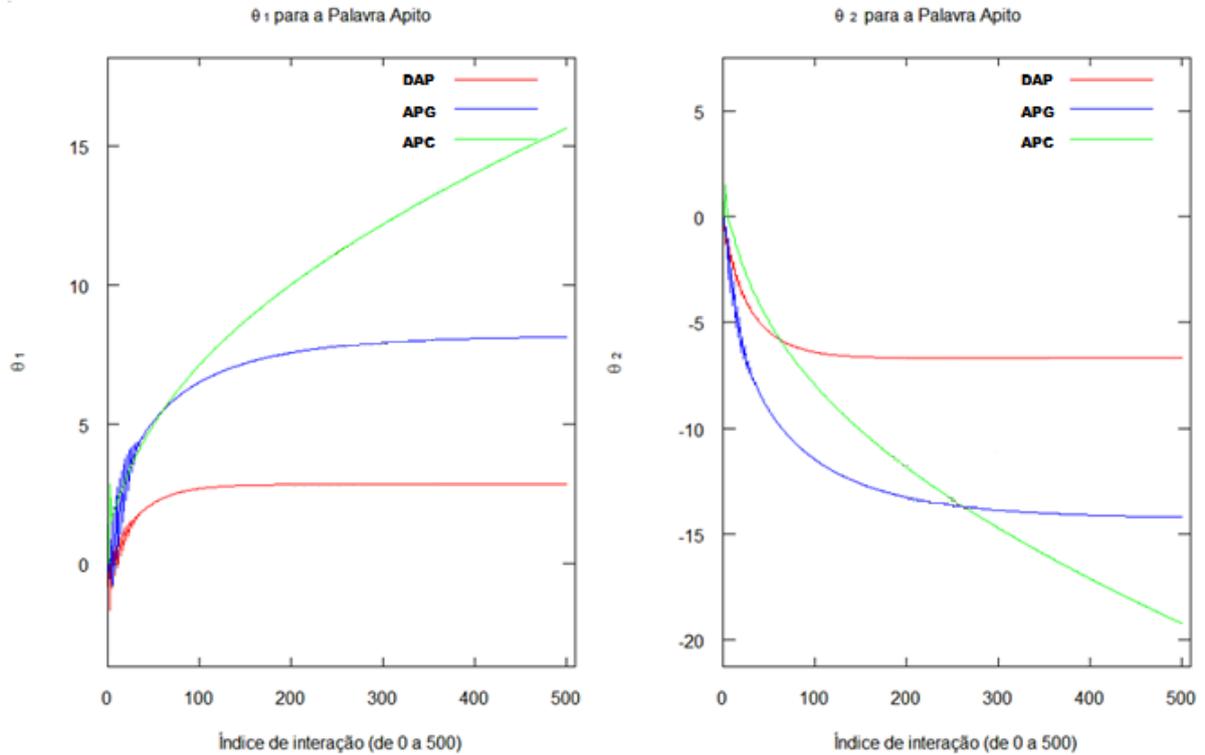
No Gráfico 5 há a convergência dos valores de θ_1 e θ_2 para os de déficit de aprendizado e aprendizado consolidado e gradativo.

Gráfico 5 – Convergência do parâmetro de aprendizado θ das três simulações para a palavra Tatu.



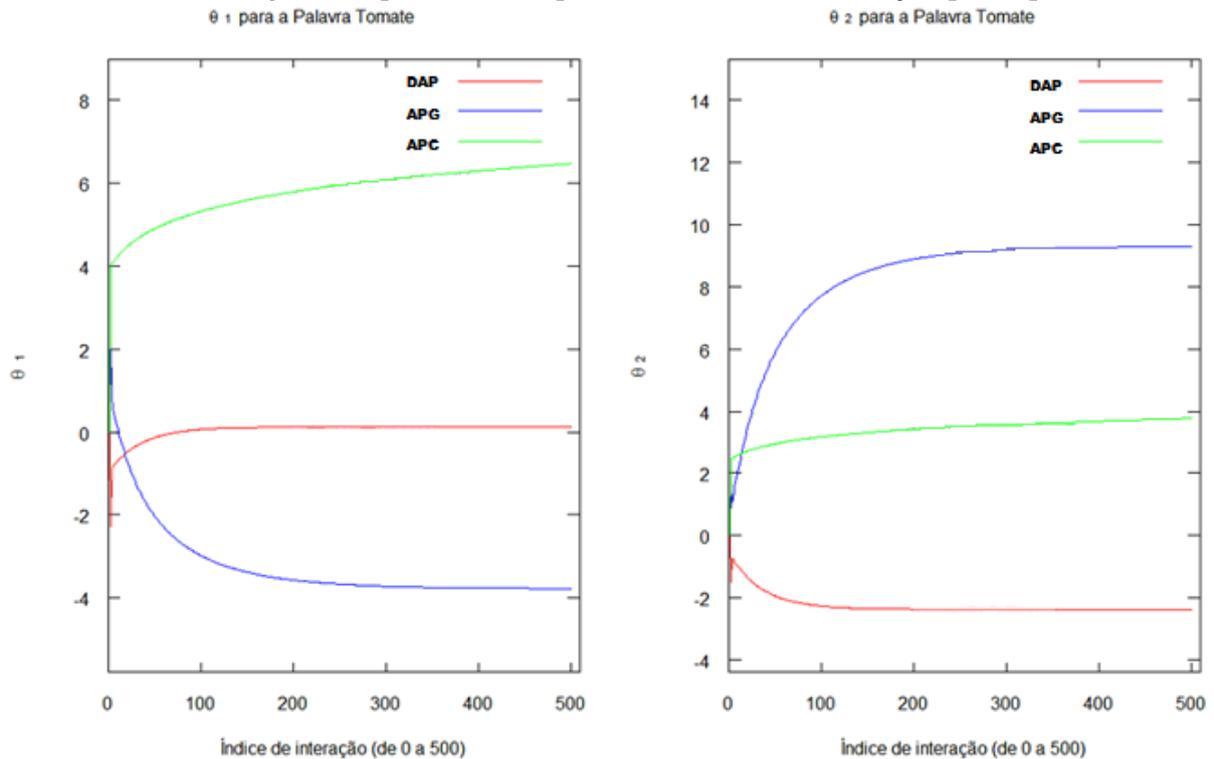
No Gráfico 6 há a convergência dos valores de θ_1 e θ_2 para os comportamentos de déficit de aprendizado e aprendizado gradativo. Para o aprendizado consolidado não há uma convergência significativa. A execução de mais interações e de uma quantidade de dados de treino maior poderão produzir resultados melhores para esse tipo de comportamento. A não convergência não acarreta consequências graves visto que um novo treinamento com novos valores poderão gerar novos resultados.

Gráfico 6 – Convergência do parâmetro de aprendizado θ das três simulações para a palavra Apito.



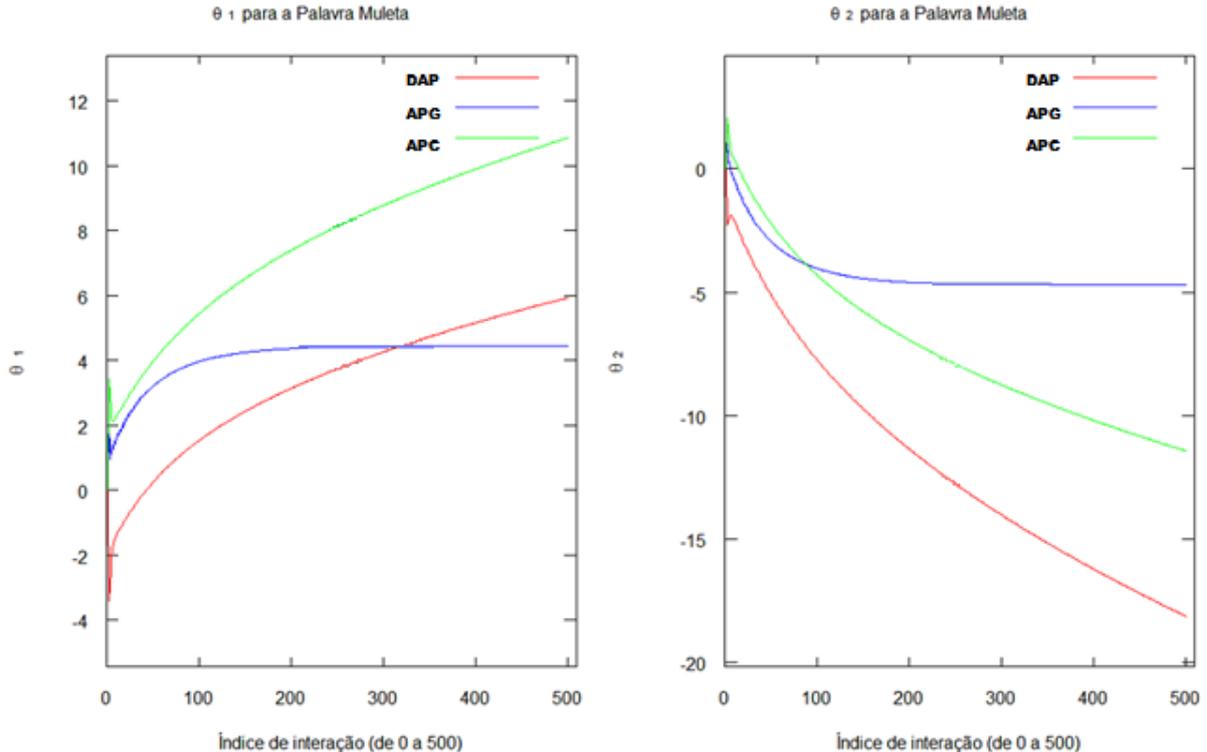
No Gráfico 7 a execução do algoritmo de aprendizado para os valores de θ_1 e θ_2 resultou na convergência dos parâmetros de aprendizado dos três comportamentos.

Gráfico 7 – Convergência do parâmetro de aprendizado θ das três simulações para a palavra Tomate



A execução do algoritmo de aprendizado para os valores de θ_1 e θ_2 no Gráfico 8 resultou na convergência dos parâmetros de aprendizado para o comportamento de aprendizado gradativo. Os comportamentos de aprendizado consolidado e déficit de aprendizado não convergiu de forma satisfatória. Uma quantidade maior de dados de treino e um maior número de interações poderiam gerar melhores resultados. A má convergência dos valores poderá gerar resultados com baixa precisão como consta no Gráfico 13.

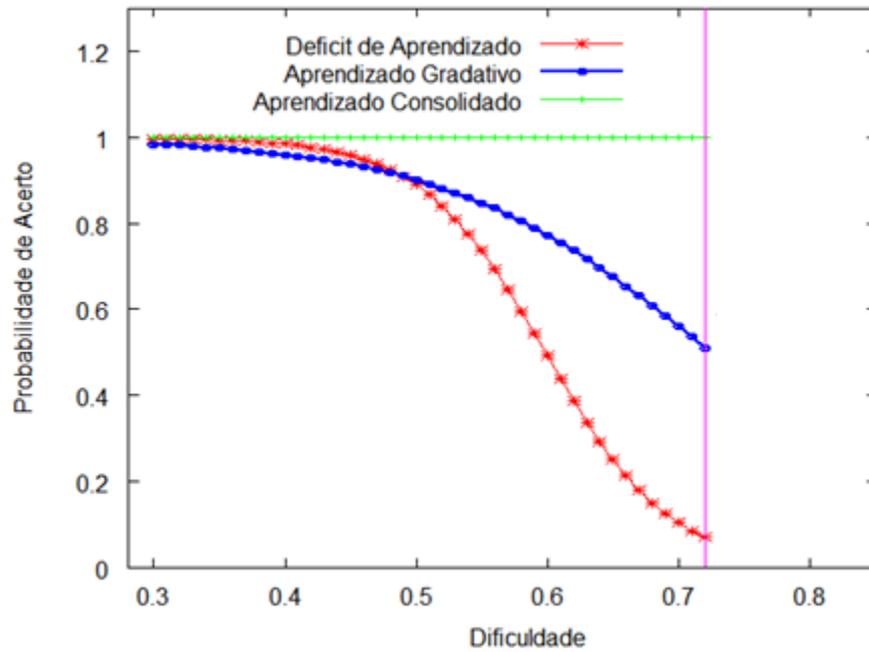
Gráfico 8 – Convergência do parâmetro de aprendizado θ das três simulações para a palavra Muleta



Com o valor de θ definido e com um valor de dificuldade de uma tarefa de ensino pode-se prever a probabilidade de acerto baseado na hipótese estabelecida. Do Gráfico 9 até o Gráfico 13 é exibido essa previsibilidade da probabilidade de acerto para os comportamentos simulados de cada palavra do experimento. Nota-se também a existência de uma linha vertical nos gráficos que correspondem ao limite de dificuldade máxima. Ela representa a tarefa mais complexa do sistema para essa palavra.

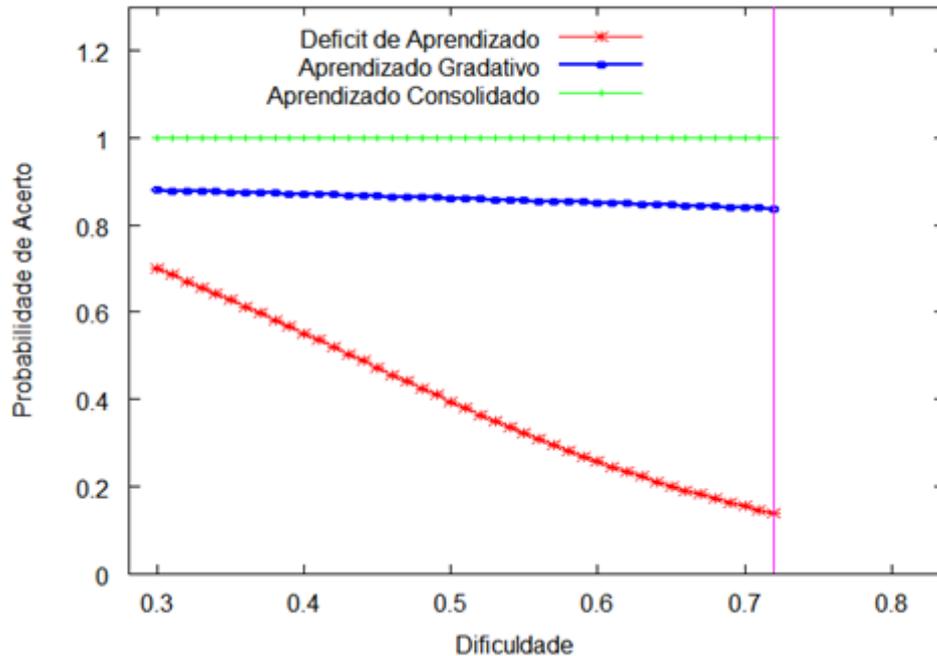
O Gráfico 9 é baseado no θ gerado referente do Gráfico 4. O valores de θ_1 e θ_2 são 11, 7012 e -19,5781 para déficit de aprendizado, 7,0302 e -9,6803 para aprendizado gradativo, 6,7440 e 3,5179 para aprendizado consolidado, respectivamente. Observa-se que quanto maior a dificuldade de uma tarefa, menor a probabilidade de acerto para essas tarefas.

Gráfico 9 – Previsão da probabilidade de acerto em um intervalo de dificuldade para a palavra Bolo.
 Valor de dificuldade pela probabilidade de acerto da palavra Bolo



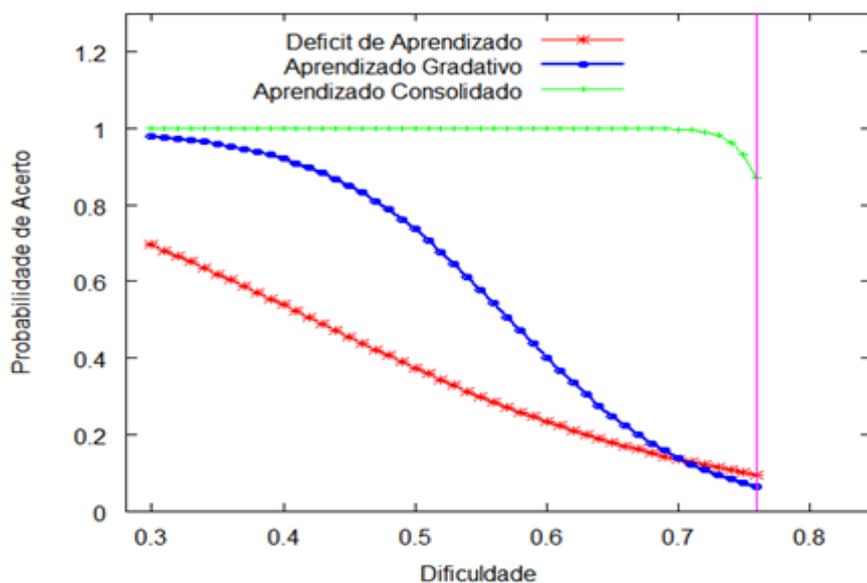
O Gráfico 10 é baseado no θ gerado referente do Gráfico 5. O valores de θ_1 e θ_2 são 2,7498 e -6,3582 para déficit de aprendizado, 2,24347 e -0,82946 para aprendizado gradativo, 6,7562 e 3,5061 para aprendizado consolidado, respectivamente. Observa-se que quanto maior a dificuldade de uma tarefa, menor a probabilidade de acerto. Nota-se também que o aprendizado consolidado se mantém em constância. Isso se deve porque para a maioria de tentativas houve bastantes acertos e pouquíssimos erros.

Gráfico 10 – Previsão da probabilidade de acerto em um intervalo de dificuldade para a palavra Tatu.
 Valor de dificuldade pela probabilidade de acerto da palavra Tatu



No Gráfico 11 o θ é baseado no Gráfico 6. O valores de θ_1 e θ_2 são 2,8374 e -6,7059 para déficit de aprendizado, 8,1261 e -14,2057 para aprendizado gradativo, 15,658 e -19,252 para aprendizado consolidado, respectivamente. Há também a mesma tendência de quanto maior a dificuldade, menor a probabilidade de acerto.

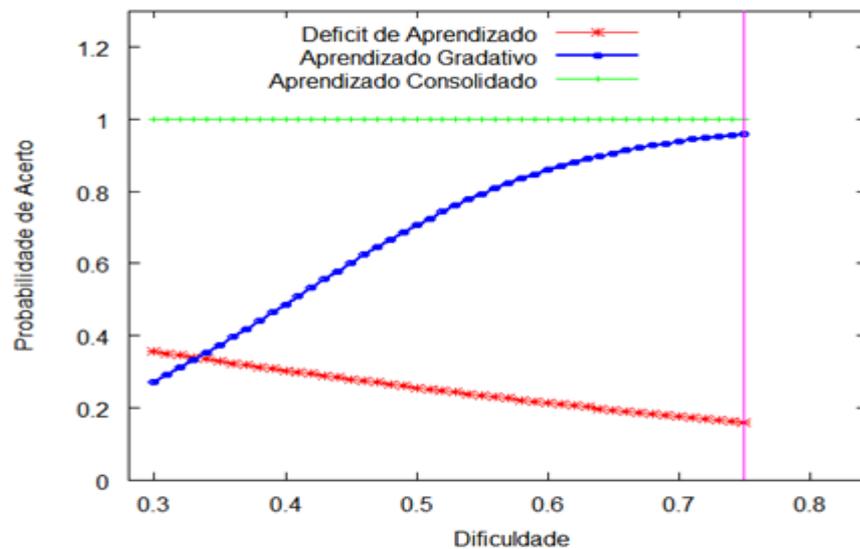
Gráfico 11 – Previsão da probabilidade de acerto em um intervalo de dificuldade para a palavra Apito.
 Valor de dificuldade pela probabilidade de acerto da palavra Apito



No Gráfico 12 o θ é baseado no Gráfico 7. O valores de θ_1 e θ_2 são 0,11749 e -2,37204 para déficit de aprendizado, -3,7689 e 9,2948 para aprendizado gradativo, 6,4596 e 3,7660 para aprendizado consolidado, respectivamente. Nesse gráfico houve uma diferença na tendência das previsões do aprendizado gradativo onde quanto maior a dificuldade maior a

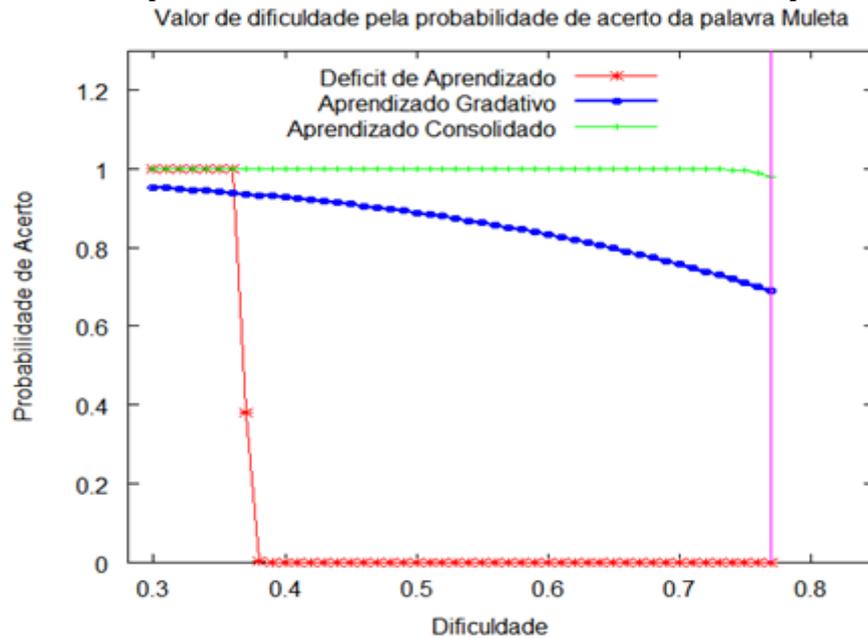
probabilidade de acerto. Isso ocorre pela elevada quantidade de acertos em tarefas consideradas complexas pelo sistema. Nesse caso é possível que exista a questão do bloqueio de aprendizado que faz com que o indivíduo possua o empecilho de acertar tarefas consideradas fáceis. Porém, após algumas tarefas de ensino apresentadas, há o entendimento geral do conjunto de tarefas de ensino resultando em acertos frequentes a partir de um dado momento.

Gráfico 12 – Previsão da probabilidade de acerto em um intervalo de dificuldade para a palavra Tomate.
Valor de dificuldade pela probabilidade de acerto da palavra Tomate



O Gráfico 13 é baseado no θ gerado referente do Gráfico 8. O valores de θ_1 e θ_2 são 5,9395 e -18,1468 para déficit de aprendizado, 4,4423 e -4,7203 para aprendizado gradativo, 10,875 e -11,445 para aprendizado consolidado, respectivamente. Observa-se que quanto maior a dificuldade de uma tarefa, menor a probabilidade de acerto. Nota-se que no comportamento de déficit de aprendizado houve uma queda brusca em sua curva. Isso mostra que em algum ponto do teste existiu uma bloqueio inerente em tentativas com dificuldades entre 0,3 e 0,4 indicando grande quantidade de tarefas com erros a partir de tarefas com dificuldade igual a 0,4 em diante.

Gráfico 13 – Previsão da probabilidade de acerto em um intervalo de dificuldade para a palavra Muleta.



Para os resultados expressos nos gráficos anteriores, a máquina de aprendizado foi executada para análise do aprendizado global do aluno, independente do tipo de tarefas referentes à leitura ou escrita. Para a análise com o aprendizado da leitura e escrita separadamente, a máquina de aprendizado classificou as palavras conforme a Tabela 20 confirmando as simulações feitas pelos psicólogos, em especial, o comportamento déficit de aprendizado. Constatou-se que o aluno com esse comportamento possui conhecimento de leitura mais não possui domínio de escrita.

Tabela 20 – Tabela com as classificações do aprendizado geradas pela MA de cada palavra das simulações realizadas. HA: Houve Aprendizado, SA: Sem Aprendizado.

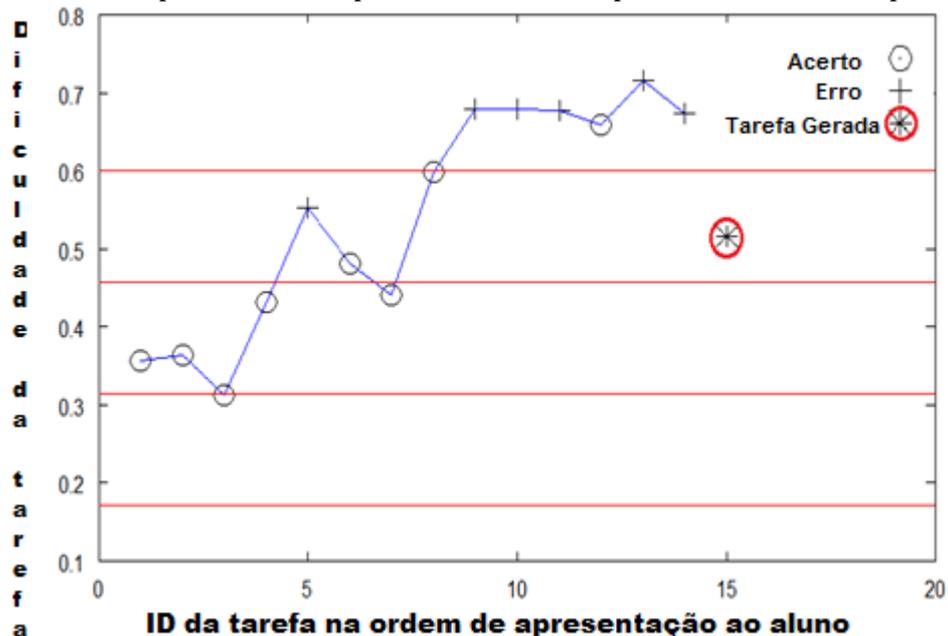
	Comportamento Simulado					
	Déficit de Aprendizado		Aprendizado Gradativo		Aprendizado Consolidado	
	Leitura	Escrita	Leitura	Escrita	Leitura	Escrita
Bolo	HA	SA	HA	HA	HA	HA
Tatu	HA	SA	HA	HA	HA	HA
Apito	HA	SA	HA	HA	HA	HA
Tomate	HA	SA	HA	HA	HA	HA
Muleta	HA	SA	HA	HA	HA	HA

6.3.3. Resultado sobre a geração de tarefas de ensino

O terceiro resultado é relacionado com o questionário enviado pelos psicólogos para a parte que retrata a geração de tarefas de aprendizado. Vale ressaltar que este resultado é parte integrante do trabalho de Pereira (2013) onde se pode verificar com maior detalhamento as características deste experimento.

As simulações geraram gráficos para cada palavra nos três comportamentos. Para ilustração é exibido o Gráfico 14 com as tarefas para o ensino da palavra “bolo”, tanto para tarefas de escrita quanto leitura. Esse gráfico refere-se a simulação do comportamento de déficit de aprendizado. Cada tarefa possui uma representação de acerto ou erro como na legenda. Há também uma tarefa que foi gerada para o aluno com dificuldade reduzida, visto a necessidade do aluno de realizar tarefas mais fáceis já que em tarefas mais complexas ele obteve fracasso.

Gráfico 14 - Tarefas para o ensino da palavra “bolo” no comportamento Déficit de Aprendizado.



Após a coleta dos resultados dos questionários houve uma concentração dos dados e uma correlação de valores de dificuldade de cada tarefa gerada pelo sistema inteligente para um valor que reflete o nível de dificuldade da tarefa, aconselhado pelos psicólogos na escala adotada (muito fácil=1, fácil=2, regular=3, difícil=4, muito difícil=5). Na Tabela 21 encontram-se os dados coletados com as opiniões dos psicólogos sobre o nível de dificuldade da tarefa escolhida como adequada para o aluno, baseado nas tarefas realizadas anteriormente, e o nível de dificuldade gerado pelo sistema. Os dados estão agrupados por grupo de alunos obtidos a partir da simulação proposta. Foi realizada uma média simples nos níveis de

dificuldade propostos pelos psicólogos e fixados na Tabela 21. Quanto mais próximo o valor gerado pelo sistema inteligente para o valor sugerido pelos especialistas, melhor o resultado.

Tabela 21 – Nível de dificuldade das tarefas geradas dado a dificuldade gerada pelo sistema inteligente e a tarefa escolhida pelos psicólogos agrupados por comportamento do aluno

	Déficit de Aprendizado		Aprendizado Gradativo		Aprendizado Consolidado	
	IA	Especialistas	IA	Especialistas	IA	Especialistas
Bolo	4	2,5	4	3,5	5	5
Tatu	4	2	4	4,5	5	5
Apito	5	2,5	5	3,5	4	3,5
Muleta	2	2	5	4,5	4	4,5

De acordo com os dados coletados com essas simulações foi concluído que o nível de dificuldade das tarefas geradas pelo sistema são próximos dos sugeridos pelos psicólogos. Os especialistas analisaram as opções de tarefas escolhendo quais destas atendem as necessidades dos estudantes. Em relação às tarefas geradas pelo sistema, que estavam ocultas entre as outras tarefas do questionário, 25% foram consideradas ideais, 41,66% foram considerados satisfatórios e 33,34% foram considerados longe do ideal. Desta forma os resultados apontam que se tem um aproveitamento de 66,66% das tarefas geradas.

7. Conclusões

A análise comportamental é um ramo da psicologia que possui grandes possibilidades de integração com técnicas de inteligência artificial. O aprendizado de máquina é uma delas. Para questões de aprendizado humano o aprendizado de máquina tem grande adaptabilidade para novas pesquisas e inovações. Aliás, o aprendizado humano deveria ser tratado de forma individualizada e personalizada visto as complexidades inerentes de cada pessoa. Esse trabalho buscou agregar várias áreas afins com a meta final de auxiliar alunos com dificuldade de aprendizado da leitura e escrita. Para isso foi necessário englobar o fator lúdico dos jogos digitais, o fator psicológico por parte dos analistas do comportamento com sua metodologia de ensino, e o fator computacional inteligente para auxiliar nessa análise dos especialistas.

De acordo com os resultados apresentados no capítulo anterior, a máquina de aprendizado se mostrou altamente pertinente em ser utilizada para prever e classificar comportamentos no programa de ensino. Há a classificação do aprendizado do aluno em palavras do programa de ensino e a previsão de acerto ou erro das tarefas de ensino. Em suma, pode-se classificar o aprendizado do aluno, se ele sabe escrever ou ler e o grau do seu aprendizado, ou seja, o sistema pode ser utilizado por pedagogos e professores para avaliar os alunos através do jogo. Apesar da MA ser utilizada em um jogo, nada impede de ser utilizada acoplada a outros programas como o GEIC.

Portanto, a contribuição desse trabalho se estende para as áreas interdisciplinares já citadas aplicado na construção de um jogo, o que consolida uma área importante de estudo na UFPA. Trata-se de um estudo relevante firmado entre computação e psicologia para a melhoria do ensino da alfabetização no Brasil.

7.1. Publicações

Este projeto de pesquisa gerou algumas publicações que valem o registro. Cada artigo foi alocados em focos específicos e divididos nas áreas multidisciplinares citadas onde um foi em uma revista (Nerino & Pereira *et al.*, 2012), outra em evento direcionado à jogos digitais (Nerino & Pereira *et al.*, 2012), uma em evento internacional direcionado a inteligência artificial (Pereira & Nerino *et al.*, 2012) e um capítulo de livro pela editora *Springer* (Pereira & Nerino *et al.*, 2012). A Tabela 23 mostra as publicações realizadas.

Tabela 22 – Artigos publicados relacionados com este projeto de pesquisa.

Nome do Artigo	Evento de Publicação e Local	Data
Máquina de aprendizagem como ferramenta de auxílio na análise comportamental no ensino da leitura	XIX Ciclo de Palestras sobre Novas Tecnologias na Educação. RENOTE- Revista sobre Novas Tecnologias na Educação, UFRGS.	Realizado em 10 a 13 de julho de 2012

A Fuzzy System for Educational Tasks for Children with Reading Disabilities	3º International Workshop on Combinations of Intelligent Methods and Applications (CIMA 2012), Montpellier, France.	Realizado em 28 de agosto de 2012
A AIED Game to help children with learning disabilities in literacy in the Portuguese language	Simpósio de entretenimento e jogos digitais, SBGAMES. Brasília-DF.	Realizado 2,3 e 4 de novembro de 2012
A Fuzzy System for Educational Tasks for Children with Reading and Writing Disabilities	Capítulo de livro pela editora <i>Springer</i> .	Enviado em novembro de 2012

7.2. Trabalhos Futuros

O desenvolvimento do trabalho está ligado a área de inteligência computacional aplicada a psicologia, porém os dados de treino foram simulados por especialistas e validados por psicólogos da UFSCar. Em relação à parte do sistema inteligente, pode-se citar alguns trabalhos futuros:

- Validar o jogo com testes envolvendo alunos reais com diferentes comportamentos.
- Elaborar um componente computacional inteligente para gerar pesos adaptados aos fatores de dificuldade considerando o andamento e histórico do aluno.
- Comparar o desempenho do algoritmo *Gradient Descent* com outros algoritmos de otimização para a máquina de aprendizado
- Comparar o aprendizado supervisionado com o aprendizado por reforço no contexto do programa de ensino.

Em relação a implementação e inovação do jogo identificou-se os seguintes trabalhos futuros considerando os avanços tecnológicos:

- Agregar mini-jogos adequados às tarefas de ensino com o objetivo de incrementar as formas de entretenimento e motivação no jogo.
- Adicionar novos tipos de tarefas de ensino para analisar novos tipos de aquisição de aprendizado, como por exemplo: falar uma palavra corretamente, escrever através de um dispositivo de caneta óptica.
- Gerar o projeto em dispositivos *tablets android* ampliando o público alvo para aqueles que não optarem em utilizar somente os computadores.

Este trabalho faz parte de um projeto de escopo maior, com a colaboração de equipes multidisciplinares onde é necessário a exploração das técnicas utilizadas por cada área de conhecimento. Espera-se que o produto desse projeto seja usado por psicólogos da UFSCar e em escolas de São Carlos, mas também se estenda a outros municípios do Brasil.

Referências Bibliográficas

AGUIAR, Eliane Vigneron B. As tecnologias e o ensino-aprendizagem. **USU, Rio de Janeiro**, Rio de Janeiro, 2008.

AMATE, Flávio Cezar. **Desenvolvimento de Jogos computadorizados para auxiliar a aquisição de base alfabética de crianças**. USP - Escola de Engenharia de São Carlos. São Carlos. 2007.

AMATE, Flavio Cezar; SLAETS, Annie France F; HELOISA, Amaral Dias O. Jogos Computadorizados para Aquisição da Linguagem Escrita na educação especial. **Conferência IADIS Ibero-Americana WWW/Internet**, 2003.

ANDREW, Y. Ng. **Machine Learning Class**, 2011. Disponível em: <<https://d19vezwu8euf16.cloudfront.net/ml/docs%2Fslides%2FLecture6.pdf>>. Acesso em 09 de Setembro de 2011.

ARTHUR, Samuel apud. Machine Learning Class, 2011. Disponível em: <<http://ml-class.com>>. Acesso em 3 de Novembro de 2011.

ASTAH. **astah**, 2012. Disponível em: <<http://astah.net/>>. Acesso em 22 de Novembro de 2012.

AZEVEDO, Maria Amélia; MARQUES, Maria Lucia. **Alfabetização hoje**. 5. ed. São Paulo: Cortez, 2001.

BELL, John; JOHNSON, Andy. Lighting and Shading. **Lighting and Shading**, 2005. Disponível em: <<http://www.cs.uic.edu/~jbell/CourseNotes/ComputerGraphics/LightingAndShading.html>>. Acesso em 19 de Janeiro de 2013.

BLACK & White. **LionHead Studios**, 2011. Disponível em: <<http://lionhead.com/Games/BW/Default.aspx>>. Acesso em 10 de Janeiro de 2012.

BLENDER. **Blender**, 2012. Disponível em: <<http://www.blender.org/>>. Acesso em 3 de Agosto de 2012.

BLIZZARD. Word of WarCraft, 1994. Disponível em: <<http://www.worldofwarcraft.com>>. Acesso em 2010 de outubro de 14.

BOOCH, Grady; JACOBSON, Ivar; RUMBAUGH, Jim. Unified Modeling Language Specification. **OMG**, 2000. Disponível em: <<http://www.omg.org/docs/formal/00-03-01.pdf>>. Acesso em 14 de agosto de 2011.

CARDOSO, Martins C. **Os Novos rumos da alfabetização infantil**: relatório encomendado pela Câmara dos Deputados ao painel internacional de especialistas em alfabetização infantil. São paulo: Memnon, 2005.

CARVALHO, Felipe Gomes. **Comportamento em Grupo de Personagens do Tipo Black&White**. Dissertação (Mestrado em Informática). ed. Rio de Janeiro: PUC, 2004.

DEBERT, Paula; MATOS, Maria Amelia; ANDERY, Maria Amalia Pie Abib. Discriminação condicional: definições, procedimentos e dados recentes. **Revista brasileira de Análise do Comportamento**, v. II, n. 1, p. 37-52, 2006.

DU BOULAY, Benedict. What does the “AI” in AIED buy? **Artificial Intelligence in Educational Software**, Londres, p. 3/1-3/4, Junho 1998.

FACELI, Katti et al. **Inteligência Artificial - Uma Abordagem de Aprendizado de Máquina**. Rio de Janeiro: LTC, 2011.

FERREIRA, Emmanoel. GAMES E IMERSÃO: a realidade híbrita como meio de imanência virtual. **Academia.edu**, 2012. Disponível em: <http://uff.academia.edu/EmmanoelFerreira/Papers/526781/GAMES_E_IMERSAO_imanencia_virtual>. Acesso em 03 de Outubro de 2012.

FILHO, Vicente Vieira. **Revolution AI engine - desenvolvimento de um motor de inteligência artificial**. Recife: Universidade Federal de Pernambuco, 2005.

FLAUSINO, Rodrigo. Estado da Arte da Inteligência Artificial para jogos eletrônicos. **Rodrigo Flausino**, 2007. Disponível em: <<http://www.rodrigoflausino.com.br/artigos-e-tutoriais/estado-da-arte-da-inteligencia-artificial-para-jogos-eletronicos/>>. Acesso em 03 de Outubro de 2012.

FORCE, Food. WFP FoodForce The Game, 2005. Disponível em: <<http://www.food-force.com/>>. Acesso em 27 de outubro de 2011.

GALVÃO, Olavo de Faria; BARROS, Romariz da Silva. Curso de Introdução a Análise Experimental do Comportamento. **Centro de Filosofia e Ciências Humanas, Departamento de Psicologia Experimental**, 2001. Disponível em: <<http://pt.scribd.com/doc/3235127/Curso-de-Introducao-a-Analise-Experimental-do-Comportamento>>. Acesso em 23 de Dezembro de 2011.

GIMP, 2009. Disponível em: <<http://www.gimp.org>>. Acesso em 5 de outubro de 2011.

GNUOCTAVE. **OCTAVE**, 2012. Disponível em: <<http://www.gnu.org/software/octave/>>. Acesso em 2012 de Maio de 4.

GRYCZAK, Vania Gevert et al. Logistic Regression Models, Neural Networks and Support Vector Machine (SVMs) in the Credit Analysis Corporate Entities. **Revista de Ciências Exatas e Naturais**, v. 12, n. 2, p. 269-293, 2010.

GUIMARÃES, Sueli. Motivação intrínseca, extrínseca e o uso de recompensas em sala de aula. **Motivação do aluno: Contribuições da psicologia contemporânea**, 2001.

GULGEMIN, Felipe. Inteligência artificial: mitos e verdades. **Tecmundo**, 2011. Disponível em: <<http://www.tecmundo.com.br/futuro/7725-inteligencia-artificial-mitos-e-verdades.htm>>. Acesso em 3 de Outubro de 2012.

IBGE. Série: SEE31 - Reprovação por série - Ensino Fundamental de 8 e 9 anos (série nova). **Séries Estatísticas & Séries Históricas**, 2011. Disponível em: <<http://seriesestatisticas.ibge.gov.br/series.aspx?vcodigo=SEE31&sv=57&t=reprovacao-serie-ensino-fundamental-8-9>>. Acesso em 14 de janeiro de 2013.

IEEE. **Starndar Computer Dictionary: A Compilation of IEEE Standard Computer Glossaries**. IEEE. New York. 1990.

IEEE. IEEE Xplore, 2012. Disponível em: <<http://ieeexplore.ieee.org/Xplore/guesthome.jsp?reload=true>>. Acesso em 25 de Dezembro de 2011.

INKSCAPE. **Inkscape**, 2012. Disponível em: <<http://inkscape.org/>>. Acesso em 22 de Novembro de 2012.

JULIANO, Marcio Cassio. Análise dos efeitos de uma programação de ensino sobre o desempenho de estudantes em um curso superior de administração de empresas. **Revista de Educação**, v. XIII, p. 23-28, 2009.

KISHIMOTO, André. Inteligência Artificial em Jogos. **Programadores de jogos**, 2004. Disponível em: <http://www.programadoresdejogos.com/trab_academicos/andre_kishimoto.pdf>. Acesso em 24 de Dezembro de 2011.

LOBÃO, Alexandre Santos et al. **XNA 3.0 para desenvolvimento de jogos no Windows, Zune e XBOX 360**. 1. ed. São Paulo: Brasport, 2010.

LYYTINEN, Heikki et al. In search of a science-based application: A learning tool for reading. **Scandinavian Journal of Psychology**, 2009. 668-675.

MACHADO, Alex F. V.; CLUA, Esteban W.; ZADROZNY, Bianca. A Method for Generating Emergent Behaviors using Machine Learning to Strategy Games. **IX SBGames**, Florianópolis-SC, Novembro 2010. 17-23.

- MAHLMANN, Tobias et al. Predicting Player Behavior in Tomb Raider: Underworld. **Computational Intelligence and Games (CIG)**, 2010. 178-185, 18-21.
- MARQUES, Leonardo Brandão; GOLFETO, Raquel Melo; DE MELO, Raquel Maria. **Manual do Usuário de Programas de Ensino via GEIC - Volume 1: Aprendendo a Ler e Escrever em Pequenos Passos**. São Carlos: [s.n.], v. 1, 2011.
- MATLAB. MathWorks Accelerating the pace of engineering and science. **MATLAB The Language of Technical Computing**, 1994. Disponível em: <<http://www.mathworks.com/products/matlab/>>. Acesso em 19 de Janeiro de 2013.
- MITCHELL, Tom. **Machine Learning**. Boston: McGraw-Hill, 1997.
- MITCHELL, Tom N. **The Discipline of Machine Learning**. School of Computer Science. Pittsburgh: Carnegie Mellon University. 2006.
- MIZUKAMI, Maria da Graça Nicoletti. **ENSINO: As abordagens do processo**. São Paulo: EPU, 1986.
- MONODEVELOP3.0. **MonoDevelop**, 2012. Disponível em: <<http://monodevelop.com/>>. Acesso em 22 de Novembro de 2012.
- NAOE, Aline. Analfabetismo no Brasil evidencia desigualdades sociais históricas. **Com Ciência - REVISTA ELETRÔNICA DE JORNALISMO CIENTÍFICO**, 2012. Disponível em: <<http://www.comciencia.br/comciencia/?section=8&edicao=74&id=923>>. Acesso em 14 de janeiro de 2013.
- NERINO, Gilberto Junior et al. **A AIED Game to help children with learning disabilities in literacy in the Portuguese language**. SBGames. Brasília: [s.n.]. 2012.
- NERINO, Gilberto Junior et al. Máquina de aprendizagem como ferramenta de auxílio na análise comportamental no ensino da leitura. **RENOTE - Revista Sobre Novas Tecnologias na Educação**, Porto Alegre, n. 10, Junho 2012.
- ORLANDO, Alex Fernando. **Uma Infra-estrutura computacional para o gerenciamento de programas de ensino individualizados**. Dissertação de Mestrado. São Carlos: UFSCar, 2009.
- PEREIRA, Adalberto Bosco Castro. **Um Sistema Fuzzy para Geração de tarefas de ensino de leitura e escrita em um jogo digital**. Belém: Dissertação (Mestrado) - Universidade Federal do Pará, Instituto de Ciências Exatas e Naturais, 2013.
- PEREIRA, Adalberto Bosco et al. **A Fuzzy System for Educational Tasks for Children with Reading and Writing Disabilities**. Montpellier : [s.n.], 2012.
- PEREIRA, Adalberto Bosco et al. **A Fuzzy System for Educational Tasks for Children with Reading Disabilities**. The 18th European Conference on Artificial Intelligence,

Workshop CIMA - 3rd International Workshop on Combinations of Intelligent Methods and Applications. Montpellier France: [s.n.]. 2012.

PEREZ, Renan. **SIGAMÍDIA**, 2011. Disponível em: <<http://www.sigamidia.com.br/gamification-nao-e-game-e-a-tendencia-do-momento/>>. Acesso em 4 de novembro de 2011.

PRADO, Maycon Rocha S et al. Jogos Digitais: definições, classificações e avaliação. **Tópicos em Engenharia de Computação VI**, 2010. Disponível em: <<http://www.dca.fee.unicamp.br/~martino/disciplinas/ia369/trabalhos/t1g1.pdf>>. Acesso em 21 de Outubro de 2011.

PRENSKY, Mark. **Digital Game-Based Learning**. [S.l.]: Paragon House Publishers, 2006.

PRESSMAN, Roger. **Engenharia de Software**. 6ª Edição. ed. [S.l.]: McGrawHill, 2006.

PROKEIN, Reiner. Free GameGraphics, 2008. Disponível em: <<http://www.reinerstilesets.de/2d-grafiken/>>. Acesso em 10 de dezembro de 2010.

PUZENAT, Didier; VERLUT, Isabelle. Behavior Analysis through Games Using Artificial Neural Networks. **Advances in Computer-Human Interactions**, 10-15 Fevereiro 2010. 134-138.

RAUBER, Thomas W. Redes Neurais Artificiais. **Thomas W. Rauber**, 2012. Disponível em: <<http://www.inf.ufes.br/~thomas>>. Acesso em 05 de Maio de 2012.

REIS, T. S; SOUZA, D. G; ROSE, J. C. Avaliação de um programa para o ensino de leitura e escrita. **Estudos em Avaliação Educacional**, v. 20, p. 425-450, 2009.

RICHARDSON, Ulla; LATVALA, Juha Matti. What is Graphogame? **Grapho Learning Initiative**. Disponível em: <<http://grapholearning.info/graphogame/whatis>>. Acesso em 20 de novembro de 2012.

ROCHA, Maycon Prado et al. Introdução aos Jogos Digitais, 2010. Disponível em: <<http://www.dca.fee.unicamp.br/~martino/disciplinas/ia369/trabalhos/t1g1.pdf>>. Acesso em 21 de outubro de 2011.

RODRIGUES, Marina da Silveira. Estilos de Aprendizagem. **espaco-praxis-pedagogicas**, 2008. Disponível em: <<http://www.educacao.salvador.ba.gov.br/site/documentos/espaco-virtual/espaco-praxis-pedagogicas/ARTIGOS%20E%20TEXTOS/estilos%20de%20%20aprendizagem%20e%20inteligencias%20multiplas.pdf>>. Acesso em 15 de Novembro de 2011.

ROSE, Julio C et al. Aquisição de leitura após história de fracasso escolar: equivalência de estímulos e generalização. **Psicologia: Teoria e Pesquisa**, p. 451-69, 1989.

- RUSSEL, Stuart; NORVIG, Peter. **Inteligência artificial**. 2ª Edição. ed. Rio de Janeiro: Campus, 2004.
- SAYEG, Elisa; CONSELHO, Federal de Psicologia. **Psicologia E Informatica: Interfaces E Desafios**. [S.l.]: Ltda, 2000.
- SCHREIBER, Ian. Game Design Concepts – Level 2: Game Design/Interação e prototipagem rápida. **XUTI GAME DEVELOPMENT**, 2011. Disponível em: <<http://xuti.net/2011/05/15/game-design-concepts-level-2game-designinteracao-e-prototipagem-rapida/>>. Acesso em 23 de março de 2012.
- SELF, John. Grounded in reality: the infiltration of AI into practical educational systems. **Artificial Intelligence in Educational Software**, Londres, p. 1/1 - 1/4, junho 1998.
- SIDMAN, Murray apud; BORTOLOTTI, Renato; DE ROSE, Júlio C. Medida do Grau de Relacionamento entre Estímulos Equivalentes. **Psicologia: Reflexão e Crítica**, v. 20, p. 252-258, Fevereiro 2007.
- SIDMAN, Murray; TAILBY, Willian. Discriminação Condicional VS Emparelhamento com o Modelo: Uma Expansão do Paradigma de Teste. **REVISTA BRASILEIRA DE ANÁLISE DO COMPORTAMENTO**, v. 2, n. 1, p. 115-139, 2006.
- SIQUEIRA, Elton S et al. **A Game for Teaching Children With disability in Reading and Writin Portuguese using Voice Recognition and Kinect Sensor**. SBGames. Salvador: [s.n.]. Novembro 2011.
- SIQUEIRA, Elton Sarmanho. **ALERPG – Jogo digital como recurso complementar para ensino de crianças com Dificuldade de Aprendizagem em Leitura e Escrita**. Dissertação (Mestrado) - Programa de Pós-Graduação em Ciência da Computação. ed. Belém: Universidade Federal do Pará, Instituto de Ciências Exatas e Naturais, 2012.
- SKINNER, Burrhus Frederic. Are theories of learning necessary ? **The Psychological Review**, n. 57, p. 196-216, 1950.
- SOUZA, Deisy G; ROSE, Júlio C. Desenvolvendo programas individualizados para o ensino de leitura. **Acta Comportamentalia**, Guadalajara, v. 14, n. 1, Junho 2006.
- THOMAS, Jane. **Guide to Managerial Persuasion and Influence**. New Jersey: Pearson Prentice Hall, 2004.
- THURAU, Christian; HETTENHAUSEN, thomas; BAUCKHAGE, Christian. Classification of Team Behaviors in Sports Video Games. **Pattern Recognition**, 2006. 1188 - 1191.
- TOBALDINI, Michele. **Arquitetura Históricas no Ambiente de um Jogo de RPG**. SBGames. [S.l.]: [s.n.]. 2006.

- TODOROV, João Cláudio. Behaviorismo e Análise Experimental do Comportamento. **Universidade de Brasília**, 1982. Disponível em: <<http://professor.ucg.br/SiteDocente/admin/arquivosUpload/15372/material/Behaviorismo%20e%20a%20An%C3%A1lise%20Exp%20Cpto.pdf>>. Acesso em 21 de Janeiro de 2012.
- TORRES, Carolina Manso Soares. **Educational Digital Game for Teaching Domestic Ecology**. Universidade Tecnica de Lisboa. Lisboa, p. 7-9. 2008.
- UNITY3D Support. **Unity 3D**, 2012. Disponível em: <<http://unity3d.com/support/>>. Acesso em 3 de Agosto de 2012.
- VISUAL Studio. **Microsoft VisualStudio**, 2010. Disponível em: <<http://www.microsoft.com/visualstudio/ptb/products/visual-studio-express-products>>. Acesso em 22 de Novembro de 2012.
- W3C. World Wide Web Consortium: Web services, 2006. Disponível em: <<http://www.w3.org/2002/ws/>>. Acesso em 20 de novembro de 2010.
- WEKA. **WEKA Documentation**, 2012. Disponível em: <<http://www.cs.waikato.ac.nz/ml/weka/>>. Acesso em 10 de Janeiro de 2012.

Anexos

Esta seção tem como objetivo mostrar documentos e informações complementares a esta pesquisa. Todos os documentos desta seção não foram da autoria do autor da presente dissertação

Anexo A. Tabela Proximidade das Palavras.

O anexo A corresponde aos dados coletados e tabelados por 4 psicólogos da UFSCar com o objetivo de identificar um grau numérico de proximidade das 15 primeiras palavras ensinadas pelo programa de ensino ALEPP. Foi sugerido para os psicólogos preencherem uma tabela indicando, no intervalo de 0 a 10, o grau de proximidade das palavras. Quanto menor o valor estipulado, menor o grau de proximidade de uma palavra em relação à outra palavra. Quanto maior o valor estipulado, maior o grau de proximidade, ou seja, mais idêntica é uma palavra em relação à outra. A Tabela 23 representa a compilação das tabelas enviadas para os psicólogos e formatadas com o nível percentual correspondente.

Tabela 23 – Indicação da Proximidade das Palavras. Número de palavras: 15. Combinações: 105. Em valor crescente de proximidade, sendo 0 = nada próximas até 1 = idênticas.

	bolo	tatu	vaca	bico	mala	tubo	pipa	cavalo	apito	luva	tomate	vovô	muleta	fita	pato
bolo	1	0,15	0,15	0,525	0,25	0,525	0,15	0,5	0,2	0,175	0,225	0,3	0,225	0,15	0,25
tatu	0,15	1	0,225	0,15	0,225	0,525	0,2	0,225	0,2	0,175	0,35	0,15	0,5	0,625	0,4
vaca	0,15	0,225	1	0,225	0,45	0,15	0,225	0,6	0,15	0,525	0,2	0,3	0,2	0,25	0,225
bico	0,525	0,15	0,225	1	0,15	0,325	0,275	0,225	0,325	0,15	0,15	0,2	0,15	0,225	0,25
mala	0,225	0,225	0,5	0,15	1	0,15	0,25	0,325	0,15	0,3	0,35	0,15	0,4	0,225	0,25
tubo	0,525	0,525	0,15	0,325	0,15	1	0,15	0,2	0,275	0,225	0,325	0,2	0,275	0,2	0,275
pipa	0,15	0,2	0,225	0,275	0,25	0,15	1	0,2	0,525	0,225	0,15	0,15	0,2	0,35	0,525
cavalo	0,5	0,225	0,525	0,225	0,35	0,2	0,2	1	0,375	0,475	0,25	0,225	0,175	0,15	0,3
apito	0,2	0,2	0,15	0,325	0,15	0,275	0,525	0,375	1	0,15	0,475	0,15	0,275	0,25	0,575
luva	0,15	0,175	0,525	0,15	0,3	0,25	0,225	0,475	0,15	1	0,2	0,2	0,375	0,275	0,15
tomate	0,225	0,325	0,2	0,15	0,35	0,3	0,15	0,25	0,475	0,2	1	0,225	0,35	0,275	0,5
vovô	0,275	0,15	0,275	0,2	0,15	0,2	0,15	0,225	0,15	0,2	0,225	1	0,15	0,15	0,2
muleta	0,225	0,475	0,2	0,15	0,4	0,275	0,2	0,175	0,275	0,375	0,325	0,15	1	0,525	0,25
fita	0,15	0,575	0,25	0,225	0,225	0,2	0,4	0,15	0,25	0,275	0,275	0,15	0,525	1	0,375
pato	0,25	0,4	0,225	0,25	0,225	0,275	0,5	0,3	0,625	0,15	0,5	0,2	0,275	0,55	1

Anexo B. Tabelas de Dados para simulação da Aquisição da Leitura

O Anexo B corresponde as simulações com autoria do psicólogo Leonardo Brandão Marques da UFSCAR. Esses dados foram retirados do banco de dados do programa GEIC de um aluno onde foi induzido erros ou acertos no atributo **Acerto**. Deste modo foi simulado o déficit de aprendizado e o aprendizado gradativo para a aquisição da leitura e analisados pela MA a fim de identificar o grau de aprendizado e se o aluno aprendeu ou não a palavra “muleta”.

Tabela 24 – Tarefas simulando o déficit de aprendizado de leitura.

Id da tentativa	Acerto	Número de Comparações	Tipo da Tentativa	Modelo	Comparação 2	Comparação 3
1	1	1	BC	muleta		
2	1	1	CB	muleta		
3	0	3	CB	muleta	luva	pato
4	0	3	AC	muleta	pato	fita
5	1	2	AC	muleta	bico	
6	1	1	CB	muleta		
7	0	2	AC	muleta	mala	
8	1	2	AC	muleta	apito	
9	1	2	AC	muleta	bolo	
10	0	2	AC	muleta	fita	
11	1	2	AC	muleta	pato	
12	0	3	AC	muleta	apito	fita
13	0	3	AC	muleta	pato	fita
14	0	3	AC	muleta	pato	fita
15	1	1	AC	muleta		
16	0	3	BC	muleta	bico	tubo
17	1	3	BC	muleta	tubo	tubo
18	1	1	CB	muleta		

Tabela 25 – Tarefas simulando o aprendizado gradativo.

Id da tentativa	Acerto	Número de Comparações	Tipo da Tentativa	Modelo	Comparação 2	Comparação 3
1	0	1	BC	muleta		
2	0	1	CB	muleta		
3	0	3	CB	muleta	luva	pato
4	1	3	AC	muleta	pato	fita
5	0	2	AC	muleta	bico	
6	0	1	CB	muleta		
7	1	2	AC	muleta	mala	
8	0	2	AC	muleta	apito	
9	1	2	AC	muleta	bolo	
10	0	2	AC	muleta	fita	
11	1	2	AC	muleta	pato	
12	0	3	AC	muleta	apito	fita

13	1	3	AC	muleta	pato	fita
14	1	3	AC	muleta	pato	fita
15	1	1	AC	muleta		
16	1	3	BC	muleta	bico	tubo
17	1	3	BC	muleta	tubo	tubo
18	1	1	CB	muleta		

Anexo C. Tabela de Dados para simulação de comportamentos.

O Anexo C corresponde as simulações tabeladas com autoria do psicólogo Leonardo Brandão Marques da UFSCar juntamente com os pesquisadores do LAAI para: Déficit de aprendizado, Aprendizado Gradativo e Aprendizado consolidado. Esses dados serviram de base para os testes comportamentais da IA, tanto para a MA quanto para o Sistema *Fuzzy*, a fim de identificar o aprendizado do aluno e gerar novas tarefas adaptadas.

Tabela 26 – Tabela com as tarefas do pré-teste para simulação dos comportamentos e análise pelo sistema inteligente. A resposta dos acertos e erros de cada tarefa estão na Tabela 27.

Id da tentativa	Data e Hora	Mini Jogo	Latência	Dificuldade	Acerto	Número de Escolhas	Tipo de Tentativa	Modelo	Escolha 1	Escolha 2	Escolha 3	Nível Dificuldade
1	--	Cubo	0	0,35625	--	2	AB	bolo	tubo			Regular
2	--	Cubo	0	0,325	--	2	AB	tatu	pato			Regular
3	--	Cubo	0	0,2625	--	2	AB	apito	luva			Muito Fácil
4	--	Cubo	0	0,428125	--	3	AB	tomate	muleta	apito		Difícil
5	--	Cubo	0	0,38125	--	3	AB	muleta	cavalo	apito		Regular
6	--	Cubo	0	0,3625	--	3	AB	bolo	vaca	fita		Regular
7	--	Cubo	0	0,365625	--	3	AB	tatu	luva	bolo		Regular
8	--	Cubo	0	0,5125	--	4	AB	apito	tomate	cavalo	bolo	Difícil
9	--	Cubo	0	0,5041667	--	4	AB	tomate	fita	muleta	tatu	Difícil
10	--	Cubo	0	0,5083333	--	4	AB	muleta	mala	tomate	apito	Difícil
11	--	Cubo	0	0,3125	--	2	AC	bolo	vaca			Regular
12	--	Cubo	0	0,375	--	2	AC	tatu	pato			Regular
13	--	Cubo	0	0,3375	--	2	AC	apito	fita			Regular
14	--	Cubo	0	0,35625	--	2	AC	tomate	tatu			Regular
15	--	Cubo	0	0,35625	--	2	AC	muleta	tomate			Regular
16	--	Cubo	0	0,43125	--	3	AC	bolo	pipa	vovô		Difícil
17	--	Cubo	0	0,425	--	3	AC	tatu	luva	mala		Regular
18	--	Cubo	0	0,48125	--	3	AC	apito	pipa	bico		Difícil
19	--	Cubo	0	0,459375	--	3	AC	tomate	muleta	tatu		Difícil
20	--	Cubo	0	0,475	--	3	AC	muleta	tubo	fita		Difícil
21	--	Cubo	0	0,5520833	--	4	AC	bolo	luva	bico	tomate	Muito Difícil
22	--	Cubo	0	0,5458333	--	4	AC	tatu	pato	mala	vaca	Difícil
23	--	Cubo	0	0,5708333	--	4	AC	apito	pato	cavalo	bolo	Difícil
24	--	Cubo	0	0,5354167	--	4	AC	tomate	cavalo	bico	tatu	Muito Difícil
25	--	Cubo	0	0,5625	--	4	AC	muleta	luva	mala	apito	Difícil
26	--	Cubo	0	0,48125	--	3	BC	bolo	vaca	cavalo		Difícil
27	--	Cubo	0	0,44375	--	3	BC	tatu	apito	bico		Difícil
28	--	Cubo	0	0,3625	--	2	BC	apito	fita			Regular

29	--	Cubo	0	0,3625	--	2	BC	tomate	cavalo			Regular
30	--	Cubo	0	0,4	--	2	BC	muleta	mala			Regular
31	--	Cubo	0	0,440625	--	3	CB	bolo	cavalo	tomate		Difícil
32	--	Cubo	0	0,434375	--	3	CB	tatu	luva	muleta		Regular
33	--	Cubo	0	0,54375	--	4	CB	apito	tubo	bico	pipa	Difícil
34	--	Cubo	0	0,53125	--	4	CB	tomate	apito	pipa	muleta	Difícil
35	--	Cubo	0	0,5541667	--	4	CB	muleta	mala	fita	tomate	Difícil
36	--	Cubo	0	0,5985714	--	2	CE	bolo	vaca			Muito Difícil
37	--	Cubo	0	0,6128572	--	2	CE	tatu	apito			Muito Difícil
38	--	Cubo	0	0,6985714	--	2	CE	apito	mala			Muito Difícil
39	--	Cubo	0	0,7557143	--	3	CE	tomate	muleta	cavalo		Muito Difícil
40	--	Cubo	0	0,7842857	--	4	CE	muleta	pato	tomate	apito	Muito Difícil
41	--	Cubo	0	0,6785714	--	2	BE	bolo	vovô			Muito Difícil
42	--	Cubo	0	0,6785714	--	2	BE	tatu	luva			Muito Difícil
43	--	Cubo	0	0,7785714	--	2	BE	apito	fita			Muito Difícil
44	--	Cubo	0	0,7785714	--	2	BE	tomate	vaca			Muito Difícil
45	--	Cubo	0	0,7928572	--	2	BE	muleta	apito			Muito Difícil
46	--	Cubo	0	0,6785714	--	2	BE	bolo	mala			Muito Difícil
47	--	Cubo	0	0,6785714	--	2	BE	tatu	vaca			Muito Difícil
48	--	Cubo	0	0,7196429	--	3	BE	apito	tubo	pato		Muito Difícil
49	--	Cubo	0	0,8071429	--	3	BE	tomate	vovô	bolo		Muito Difícil
50	--	Cubo	0	0,8071429	--	3	BE	muleta	luva	pipa		Muito Difícil
51	--	Cubo	0	0,677381	--	4	BE	bolo	vaca	bico	tubo	Muito Difícil
52	--	Cubo	0	0,677381	--	4	BE	tatu	tubo	mala	pipa	Muito Difícil
53	--	Cubo	0	0,8205357	--	4	BE	apito	pato	cavalo	muleta	Muito Difícil
54	--	Cubo	0	0,7928572	--	2	BE	tomate	muleta			Muito Difícil
55	--	Cubo	0	0,7928572	--	2	BE	muleta	tomate			Muito Difícil
56	--	Cubo	0	0,6585714	--	2	AE	bolo	pato			Muito Difícil
57	--	Cubo	0	0,6585714	--	2	AE	tatu	luva			Muito Difícil
58	--	Cubo	0	0,7585714	--	2	AE	apito	bico			Muito Difícil
59	--	Cubo	0	0,7871429	--	3	AE	tomate	luva	bico		Muito Difícil
60	--	Cubo	0	0,8157143	--	4	AE	muleta	vovô	mala	bolo	Muito Difícil

61	--	Cubo	0	0,7157143	--	4	AE	bolo	bico	pipa	vovô	Muito Difícil
62	--	Cubo	0	0,6728572	--	2	AE	tatu	apito			Muito Difícil
63	--	Cubo	0	0,7585714	--	2	AE	apito	fita			Muito Difícil
64	--	Cubo	0	0,7585714	--	2	AE	tomate	vovô			Muito Difícil
65	--	Cubo	0	0,7585714	--	2	AE	muleta	tubo			Muito Difícil
66	--	Cubo	0	0,6728572	--	2	AE	bolo	tomate			Muito Difícil
67	--	Cubo	0	0,6585714	--	2	AE	tatu	pato			Muito Difícil
68	--	Cubo	0	0,7871429	--	3	AE	apito	tatu	bico		Muito Difícil
69	--	Cubo	0	0,78	--	4	AE	tomate	vaca	bolo	apito	Muito Difícil
70	--	Cubo	0	0,78	--	4	AE	muleta	apito	vovô	tatu	Muito Difícil

Tabela 27 – Acertos dos comportamentos simulados da Tabela 26 anterior.

Id da tentativa	Acerto (Para Déficit de Aprendizado)	Acerto (Para Aprendizado Gradativo)	Acerto (Para Aprendizado Consolidado)
1	1	1	1
2	1	1	1
3	1	1	1
4	1	1	1
5	0	1	1
6	1	1	1
7	1	1	1
8	1	1	1
9	1	1	1
10	0	1	1
11	1	1	1
12	1	1	1
13	0	1	1
14	0	1	1
15	1	1	1
16	1	1	1
17	0	0	1
18	0	1	1
19	0	0	1
20	0	1	1
21	0	0	1
22	0	1	1
23	0	0	1
24	0	0	1
25	0	0	1
26	1	1	1
27	0	1	1
28	1	1	1
29	0	0	1
30	0	1	1
31	1	1	1
32	0	1	1
33	0	0	1
34	0	1	1
35	0	1	1
36	1	1	1
37	1	1	1
38	1	1	1
39	1	1	1
40	0	1	1
41	0	1	1
42	0	1	1
43	0	0	1
44	0	1	1
45	0	0	1
46	0	1	1
47	1	0	1
48	0	0	1
49	0	1	1
50	0	0	1
51	0	1	1
52	0	1	1
53	0	0	0

54	0	1	1
55	0	1	1
56	1	1	1
57	0	1	1
58	0	0	1
59	0	1	1
60	0	1	0
61	0	0	1
62	0	1	1
63	0	0	1
64	0	1	1
65	0	1	1
66	0	0	1
67	0	1	1
68	0	0	0
69	0	1	1
70	0	1	1

Apêndices

Esta seção tem como objetivo mostrar documentos e informações acabados desta pesquisa. Todos os documentos desta seção foram da autoria do autor da presente dissertação e o mesmo deixa o livre acesso de tais informações, de tal forma que seu nome seja citado como responsável pelas informações.

Apêndice A. Algoritmo de Proximidade de Palavras.

O algoritmo de proximidade de palavras tem a função de determinar o quanto uma palavra é próxima ou idêntica a partir de especificações dos psicólogos e de suas experiências no programa de ensino. Ele é utilizado para caracterizar tarefas de leitura. Seu pseudocódigo está exibido no esquema abaixo.

Parâmetros de Entrada:

- **ListaDePalavras:** Essa variável é uma lista de objetos com todas as palavras do sistema.
- **Palavra1:** Essa variável representa a primeira palavra que deseja-se comparar com outra palavra para identificar a proximidade. Essa palavra é a palavra modelo.
- **Palavra2:** Essa variável representa a segunda palavra cujo deseja-se comparar com a primeira palavra. Essa palavra é a palavra escolha para comparação com o modelo.

Variáveis Locais:

- **VetorTamanhoDasPalavras:** Esse vetor identifica o tamanho de cada palavra onde o tamanho é definido por cada letra da palavra.
- **DiferencaMaxima:** É a variável que vai identificar qual a diferença entre a maior palavra do sistema e a menor palavra do sistema.
- **TamanoDaPalavra1:** é o tamanho da Palavra1.
- **TamanoDaPalavra2:** é o tamanho da Palavra2.
- **DifEntreAsDuasPalavras:** Identifica a diferença entre a Palavra1 e a Palavra2.
- **ValorDaRegra1, 2, 3, 4 e 5:** Identifica qual o valor de proximidade que será atribuído comparando-se as duas palavras para cada regra de proximidade.
- **PesoDaRegra1, 2, 3, 4 e 5:** Identifica o peso de cada regra do sistema: Por definição dos psicólogos foi atribuído o peso dessas variáveis como : **0,2 ; 0,225 ; 0,175 ; 0,1 ; 0,3** respectivamente.

Algoritmo 3 Algoritmo Proximidade de Palavras

```

1:  Função Real AlgoritmoProximidade ( Lista ListaDePalavras , Palavra1, Palavra2 )
2:
3:  Início
4:
5:  Laço para cada palavra da ListaDePalavras faça
6:      VetorTamanhoDasPalavras.Adicionar (PegarTamanho(palavra))
7:  FimLaço
8:
9:  Ordenar(VetorTamanhoDasPalavras)
10: DiferencaMaxima=PegarUltimoRegistro(VetorTamanhoDasPalavras) -
    PegarPrimeiroRegistro(VetorTamanhoDasPalavras)
11: TamanhoDaPalavra1 = pegaTamanhoDaPalavra(Palavra1)
12: TamanhoDaPalavra2= pegaTamanhoDaPalavra(Palavra2)
13: DifEntreAsDuasPalavras = ValorAbsoluto(TamanhoDaPalavra1- TamanhoDaPalavra2)
14:
15:  //Começa Atribuição dos valores das regras
16:
17:  //Regra 1 – Diferença entre o tamanho das palavras
18:  Se ((DifEntreAsDuasPalavras) == 0) então,
19:      ValorDaRegra1= PesoDaRegra1
20:  FimSe
21:  Se ((DifEntreAsDuasPalavras) != 0) então,
22:      ValorDaRegra1 = PesoDaRegra1 – ((DifEntreAsDuasPalavras/DiferencaMaxima)*
PesoDaRegra1
23:  FimSe
24:
25:  //Regra 2 - Primeira Letra é Igual?
26:  Se ( PegarPrimeiraLetra(Palavra1) == PegarPrimeiraLetra(Palavra2)) então
27:      ValorDaRegra2 = PesoDaRegra2
28:  Senão
29:      ValorDaRegra2 = 0
30:  FimSe
31:
32:  //Regra 3 - Primeira Sílabas é Igual?
33:  Se ( PegarPrimeiraSílabas(Palavra1) == PegarPrimeiraSílabas (Palavra2)) então
34:      ValorDaRegra3 = PesoDaRegra3
35:  Senão
36:      ValorDaRegra3 = 0
37:  FimSe
38:
39:  //Regra 4 – Última Sílabas é Igual?
40:  Se ( PegarUltimaSílabas(Palavra1) == PegarUltimaSílabas (Palavra2)) então
41:      ValorDaRegra4 = PesoDaRegra4
42:  Senão

```

```
43:     ValorDaRegra4 = 0
44: FimSe
45:
46: //Regra 5 – Última Letra é Igual?
47: Se ( PegarUltimaLetra(Palavra1) == PegarUltimaLetra (Palavra2)) então
48:     ValorDaRegra5 = PesoDaRegra5
49: Senão
50:     ValorDaRegra5 = 0
51: FimSe
52:
53: Retorna ValorDaRegra1 + ValorDaRegra2+ ValorDaRegra3+ ValorDaRegra4+ ValorDaRegra5
54:
55: FimFunção
```

Apêndice B. Algoritmo: Lógica de geração de sessão de ensino.

Este apêndice contém o algoritmo em sua primeira versão que incrementa uma sessão de ensino para o aluno após a execução do pré-teste inicial. O objetivo é gerar uma sessão de ensino no qual sejam executadas tarefas para ensinar ao aluno as palavras do sistema. Palavras que já foram “aprendidas” são classificadas como sonda e eventualmente são apresentadas ao aluno.

Regras utilizadas para verificar características de aprendizado da palavra (via *Machine Learning*):

- **regra1**: Pegar a palavra que o aluno mais sabe (grau de aprendizado maior) e que ele ainda não aprendeu.
- **regra2**: Pegar a palavras que foi mais difícil de ser “aprendida”. Ou seja, foi “aprendida” mas tem o menor grau de aprendizado.

Variáveis principais.

- **ListadePalavras** = Lista com todas as palavras do Jogo.
- **ListaDeSonda** = Lista de palavras que o aluno já aprendeu.
- **ListaASerEnsinada** = Lista de todas as palavras que o aluno não aprendeu.
- **ListaAuxiliarDePalavrasASeremEnsinadas** = É uma lista auxiliar de palavras a serem ensinadas contendo um número restrito de palavras. A geração de tarefas e verificação do aprendizado serão executados para as palavras dessa lista.
- **PalavrasAprendidas** = contador para identificar quantas palavras foram “aprendidas” nesta sessão.

Algoritmo 4 Algoritmo Lógica Geração De Sessão de Ensino

```

1:  Início
2:  Execução e Fim do Pré-teste
3:  Verificar cada palavra em ListaDePalavras
4:    Se aprendeu
5:      incluir em ListaDeSonda
6:    Senão,
7:      incluir em ListaASerEnsinada
8:    fim Se
9:  fim do Verificar
10: Gerar ListaAuxiliarDePalavrasASeremEnsinadas contendo 3 palavras da regra1 da
ListaASerEnsinada
11: Excluir as 3 palavras da regra1 da ListaASerEnsinada
12: Iniciar contador PalavrasAprendidas igual a 0
13: Enquanto ListaASerEnsinada for diferente de Vazio
14:   Se PalavrasAprendidas igual a 4
15:     Salva no Base de Dados e finaliza o sessão, e aluno retorna a jogar outro dia.
16:   fim Se
17:   Se tempo decorrido for maior que 40 minutos
18:     Salva no Base de Dados e finaliza o sessão, e aluno retorna a jogar outro dia.
19:   fim Se
20:   Gerar tarefas para cada palavra da ListaAuxiliarDePalavrasASeremEnsinadas
21:   Executar as tarefas da ListaAuxiliarDePalavrasASeremEnsinadas no jogo
22:   Verificar aprendizado de cada palavra da ListaAuxiliarDePalavrasASeremEnsinadas
23:   Se aprendeu
24:     incluir palavra aprendida em ListaDeSonda
25:     excluir palavra aprendida de ListaAuxiliarDePalavrasASeremEnsinadas
26:     incluir em ListaAuxiliarDePalavrasASeremEnsinadas a palavra da regra1 ListaASerEnsinada
27:     remover de ListaASerEnsinada a palavra incluída na linha acima
28:     incrementar 1 a PalavrasAprendidas
29:   fim Se
30: fim do Verificar
31: Se ListaDeSonda diferente de vazio
32:   gerar uma tarefa respeitando a regra2 para palavras na ListaDeSonda
33:   executa a tarefa acima gerada no jogo
34:   Se aluno errar a tarefa para a palavra sonda gerada acima
35:     Incluir essa palavra em ListaASerEnsinada
36:     Retirar palavra da ListaDeSonda
37:   Fim Se
38: fim Se
39: fim do Enquanto
40: Fim (Game Over)

```

Apêndice C. Questionário para Validação do Aprendizado de Leitura

Este apêndice contém o questionário enviado para os especialistas relacionado com a avaliação da leitura na simulação de dois comportamentos. Esse questionário foi elaborado juntamente com o psicólogo Leonardo Brandão Marques da UFSCar.

Questionário para Validação de Dados.

Este questionário visa identificar como especialistas no ensino de leitura avaliam a evolução da aprendizagem de uma palavra em uma criança em fase de alfabetização. A tarefa de ensino consiste na apresentação de uma palavra (ditada ou escrita) como dica, o aluno deve selecionar dentre três alternativas disponíveis a palavra escrita ou figura correspondente com essa dica.

A simulação do desempenho nos gráficos exibe uma amostra de várias dessas tarefas para o ensino de uma palavra específica, indicando o erro e o acerto para cada uma. Para a leitura do gráfico você deve levar em consideração que:

- O eixo horizontal exibe as tentativas na ordem em que foram apresentadas ao aluno.
- O eixo vertical indica o nível de dificuldade da tarefa em cada apresentação.
- Os pontos significam as tentativas onde os círculos (“o”) significam acertos e as cruzes (“+”) significam erros.
- Para a primeira simulação foram realizadas 18 tentativas. Para a segunda simulação foram realizadas 18 tentativas.

Agora pedimos que você classifique o nível do grau de aprendizagem do aluno em cada uma das simulações conforme as considerações abaixo.



Observações:

Escala de concordância adotada para o **Grau de Aprendizagem**:

Nível do Grau de Aprendizagem da palavra **insuficiente** = **1 ponto**,

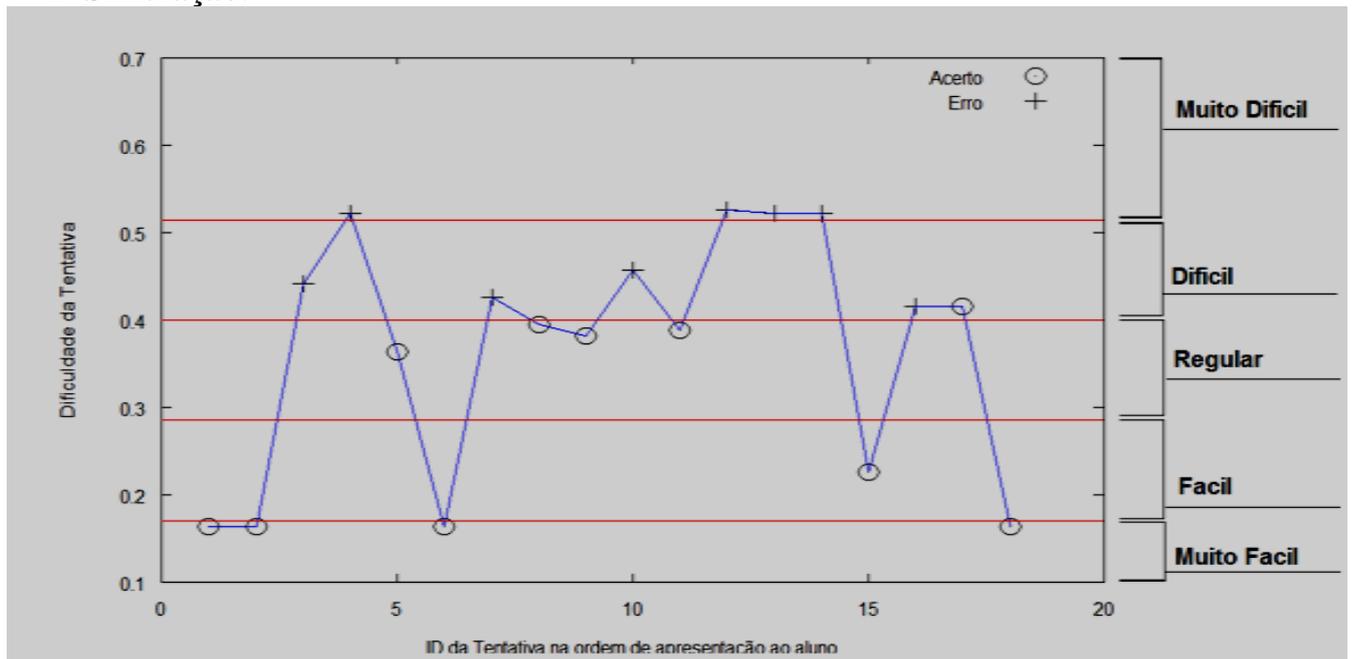
Nível do Grau de Aprendizagem da palavra **regular**= **2 pontos**,

Nível do Grau de Aprendizagem da palavra **bom** = **3 pontos**,

Nível do Grau de Aprendizagem da palavra **ótimo**= **4 pontos**,

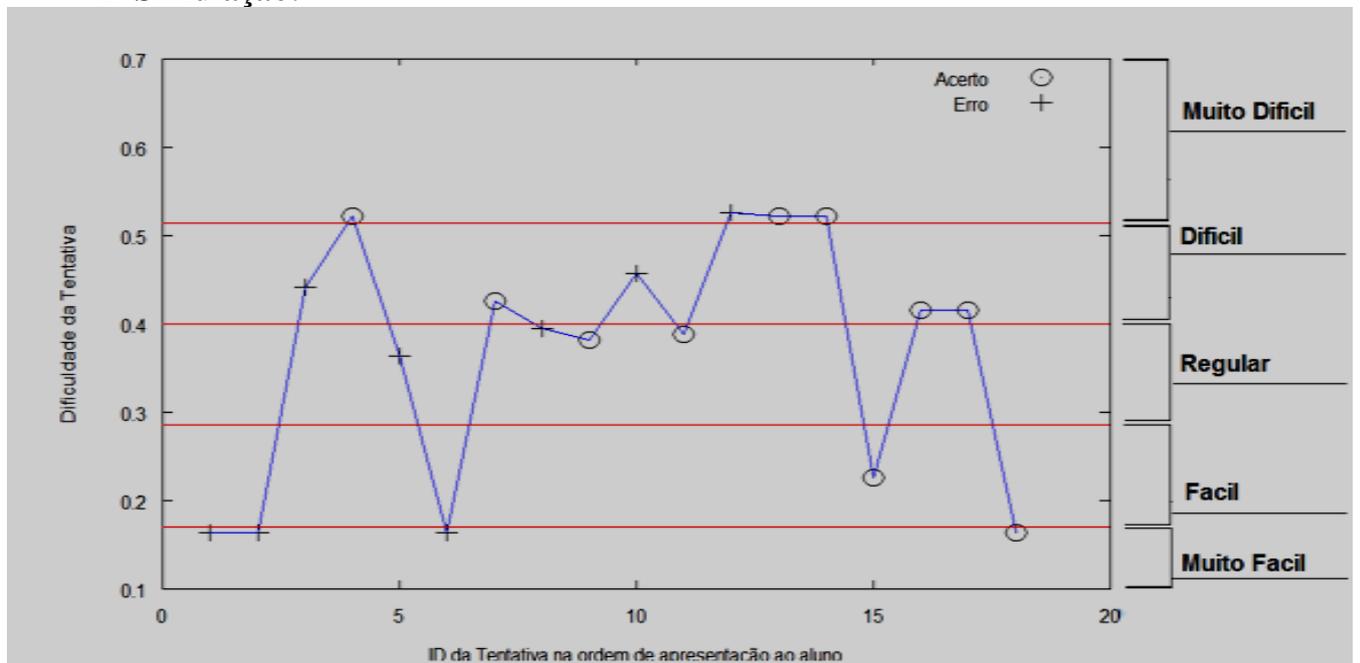
Nível do Grau de Aprendizagem da palavra **excelente**= **5 pontos**.

1ª Simulação:



1ª Simulação	
Grau de Aprendizagem na Palavra	Está aprendendo a palavra (Sim ou Não)
(preencha aqui o Grau de Aprendizagem)	(preencha aqui se ele está aprendendo ou não: Sim ou Não)

2ª Simulação:



2ª Simulação	
Grau de Aprendizagem na Palavra	Está aprendendo a palavra (Sim ou Não)
(preencha aqui o Grau de Aprendizagem)	(preencha aqui se ele está aprendendo ou não : Sim ou Não)

Obrigado desde já!

Apêndice D. Questionário para Validação da Dificuldade e Geração das Tarefas

Este apêndice contém a cópia de parte dos questionários enviados para especialistas a fim de validar as formulas de dificuldade, os níveis de dificuldade e se, baseado em simulações de comportamentos, qual tarefa seria a mais adequada para o aluno. O questionário buscou simular 3 comportamentos (Deficit de aprendizado, Aprendizado Gradativo e Aprendizado Consolidado). Para cada comportamento houve 5 modelos de referência (bolo, tatu,apito,tomate,muleta). Para cada modelo de referência houve 14 tarefas (7 tarefas de leitura e 7 de escrita). Esse questionário foi elaborado juntamente com o psicólogo Leonardo Brandão Marques da UFSCar. Para consulta, este questionário está disponível no seguinte *link* de internet: http://www.laai.ufpa.br/publicacoes/questionarios/questionario_validate_difficulty_and_generate_tasks.pdf .

Questionários

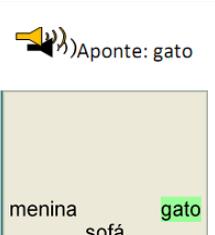
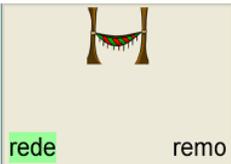
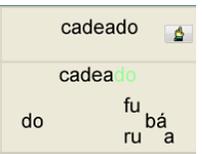
Validação da Geração e Dificuldade das tarefas de ensino.

Este questionário visa identificar como especialistas no ensino de leitura e escrita avaliam a evolução da aprendizagem de uma palavra em uma criança em fase de alfabetização. A tarefa de ensino consiste na apresentação de uma palavra, ditada ou escrita, como dica (sempre na parte superior da tela da tarefa) e o aluno deve selecionar dentre as alternativas disponíveis a palavra escrita ou figura correspondente com essa dica. Para a montagem da palavra também se encontra uma dica com o modelo correto juntamente com sílabas onde o aluno deve selecioná-las na ordem correta compondo a resposta.

Nos questionários seguintes todas as tarefas que estão em formato texto em vez de imagem são tarefas similares às imagens correspondentes. O modelo significa a escolha correta que está presente no conjunto de escolhas, como a escolha correta que o aluno deve selecionar. O modelo também é a dica de qual escolha correta ele deve selecionar. As escolhas que não são modelo significam as alternativas incorretas. Subtende-se que as escolhas incorretas estão embaralhadas na tela. Tarefas do Tipo AE, BE e CE que são tarefas de montagem das sílabas das palavras,

possuem escolhas como palavras inteiras, porém no momento da exibição na tela para o aluno, as palavras escolhidas são desmembradas fornecendo somente as sílabas nas telas de forma similar as imagens de exemplos das tarefas de montagem de palavra.

Abaixo segue uma tabela de referencia que descreve os tipos de tarefas envolvidas nesse questionário para melhor entendimento.

Tipo de Tarefa	AB	AC	BC	CB	AE	BE	CE
Ilustração da tarefa							
Função da Criança	Diante da instrução falada “Aponte aluno”, o aluno deverá selecionar a figura do ‘aluno’.	Diante da instrução falada “Aponte gato”, o aluno deverá selecionar a palavra escrita «gato».	Diante da figura de uma rede, o aluno deverá selecionar a palavra escrita «rede».	Diante da palavra escrita «vela», o aluno deverá selecionar a «vela».	Diante da instrução falada “Escreva rede”, o aluno deverá compor a palavra «rede», escolhendo as letras na ordem correta.	Diante da instrução falada “Que figura é essa?”, o aluno deverá compor a palavra «cadeado», escolhendo as letras na ordem correta.	Diante da instrução falada “Que palavra é essa?”, o aluno deverá compor a palavra «cadeado», escolhendo as letras na ordem correta.

Questionário 1: Instruções:

1. Marque um X na escala abaixo de cada tarefa uma nota (de 1 a 5) que represente o nível de dificuldade da mesma.
 a. Marque no 1 para as tarefas que você considera fáceis e marque 5 para tarefas que você considera mais difíceis.

		Tarefa Fácil					Tarefa Difícil						
		1	2	3	4	5			1	2	3	4	5
TAREFA 1	TAREFA 2	TAREFA 3	TAREFA 4	TAREFA 5	TAREFA 6	TAREFA 7							
Dificuldade: Regular	Dificuldade: Regular	Dificuldade: Regular	Dificuldade: Difícil	Dificuldade: Muito Difícil	Dificuldade: Difícil	Dificuldade: Difícil							
<p>🔊 Aponte Bolo!</p> 	<p>Tipo de Tarefa: AB Modelo: Bolo Escolha 1: Vaca Escolha 2: Fita</p>	<p>🔊 Aponte Bolo!</p> <p>Vaca Bolo</p>	<p>Tipo de Tarefa: AC Modelo: Bolo Escolha 1: Pipa Escolha 2: vovô</p>	<p>Tipo de Tarefa: AC Modelo: Bolo Escolha 1: Luva Escolha 2: Bico Escolha 3: Tomate</p>	 <p>Bolo Vaca cavalo</p>	<p>Bolo</p> 							
Acertou	Acertou	Acertou	Acertou	Errou	Acertou	Acertou							
1 2 3 4	1 2 3 4	1 2 3 4	1 2 3 4	1 2 3 4	1 2 3 4	1 2 3 4							
TAREFA 8	TAREFA 9	TAREFA 10	TAREFA 11	TAREFA 12	TAREFA 13	TAREFA 14							
Dificuldade: Muito Difícil	Dificuldade: Muito Difícil	Dificuldade: Muito Difícil	Dificuldade: Muito Difícil	Dificuldade: Muito Difícil	Dificuldade: Muito Difícil	Dificuldade: Muito Difícil							
<p>Bolo</p> <p>lo bo va ca</p>	 <p>vô bo lo vo</p>	<p>Tipo de Tarefa: BE Modelo: Bolo Escolha 1: Mala</p>	<p>Tipo de Tarefa: BE Modelo: Bolo Escolha 1: Vaca Escolha 2: Bico Escolha 3: Tubo</p>	<p>🔊 Bolo!</p> <p>to bo lo pa</p>	<p>Tipo de Tarefa: AE Modelo: Bolo Escolha 1: Pipa Escolha 2: Bico Escolha 3: Vovô</p>	<p>Tipo de Tarefa: AE Modelo: Bolo Escolha 1: Tomate</p>							
Acertou	Errou	Errou	Errou	Acertou	Errou	Errou							
1 2 3 4 5	1 2 3 4 5	1 2 3 4 5	1 2 3 4 5	1 2 3 4 5	1 2 3 4 5	1 2 3 4 5							

Observação: As imagens que tem uma figura de um alto-falante escrito “aponte ...” representam estímulos sonoros.

Agora, considerando o desempenho do aluno das tarefas acima, indique qual a tarefa que você recomenda para ser aplicada em seguida.

1. Marque um X abaixo da tarefa que represente o nível de dificuldade adequado à próxima tarefa e, para a tarefa escolhida, o nível de dificuldade.

1ª SUGESTÃO DE TAREFA	2ª SUGESTÃO DE TAREFA	3ª SUGESTÃO DE TAREFA	4ª SUGESTÃO DE TAREFA	5ª SUGESTÃO DE TAREFA
 Bolo! bo lu bo tu lo va	 vovô bolo	 vo va ca lo vô lo bo	 bolo! ta le mu la ma vô vo	 vaca bolo vovô
()	()	()	()	()
				