

**UNIVERSIDADE FEDERAL DO ESPÍRITO SANTO
DEPARTAMENTO DE INFORMÁTICA
PROGRAMA DE PÓS-GRADUAÇÃO EM INFORMÁTICA**

SABRINA SIQUEIRA PANCERI

**APOIO COMPUTACIONAL PARA A MEDIAÇÃO
PEDAGÓGICA EM “DEBATE DE TESES”**

**VITÓRIA
2016**

SABRINA SIQUEIRA PANCERI

**APOIO COMPUTACIONAL PARA A MEDIAÇÃO
PEDAGÓGICA EM “DEBATE DE TESES”**

Dissertação apresentada ao Programa de Pós-Graduação em Informática do Departamento de Informática da Universidade Federal do Espírito Santo, como requisito parcial para obtenção do título de Mestre em Informática, na área de concentração de Informática na Educação.

Orientador: Prof. Dr. Crediné Silva de Menezes.

VITÓRIA
2016

Dados Internacionais de Catalogação-na-publicação (CIP)
(Biblioteca Setorial Tecnológica,
Universidade Federal do Espírito Santo, ES, Brasil)

P188a Panceri, Sabrina Siqueira, 1985-
Apoio computacional para a mediação pedagógica em debate
de teses/ Sabrina Siqueira Panceri. – 2016.
113 f. : il.

Orientador: Crediné Silva de Menezes.

Dissertação (Mestrado em Informática) – Universidade
Federal do Espírito Santo, Centro Tecnológico.

1. Arquiteturas pedagógicas. 2. Agentes inteligentes (Software).
3. Recuperação da informação. 4. Processamento de textos
(Computação). I. Menezes, Crediné Silva de. II. Universidade
Federal do Espírito Santo. Centro Tecnológico. III. Título

CDU: 004

SABRINA SIQUEIRA PANCERI

APOIO COMPUTACIONAL PARA A MEDIAÇÃO PEDAGÓGICA EM “DEBATE DE TESES”

Dissertação apresentada ao Programa de Pós-Graduação em Informática do Departamento de Informática da Universidade Federal do Espírito Santo, como requisito parcial para obtenção do título de Mestre em Informática, na área de concentração de Informática na Educação.

Aprovada em Vitória-ES no dia 29 de abril de 2016.

COMISSÃO EXAMINADORA

Prof. Dr. Crediné Silva de Menezes
Programa de Pós-Graduação em Informática – UFES
Faculdade de Educação – UFRGS
Orientador

Prof. Dr. Davidson Cury
Programa de Pós-Graduação em Informática – UFES

Prof. Dr. José Francisco de Magalhães Netto
Programa de Pós-Graduação em Informática – UFAM

RESUMO

O advento da cultura digital trouxe novas possibilidades de interação e comunicação entre os usuários, promovendo o aumento destas trocas e também o aumento dos dados produzidos por elas. Seguindo essa nova perspectiva, as interações em ambientes virtuais de ensino e aprendizagem são registradas para possibilitar que o professor possa (re)visitar os espaços e produções de cada aluno buscando informações para auxiliá-los de diversas formas. As Arquiteturas Pedagógicas apresentam-se como uma possibilidade de organização dessas interações, em especial a Arquitetura Pedagógica “Debate de Teses” que propõe um fluxo de interações, realizadas através da produção de textos, para apoiar a construção de conhecimento daqueles que dela participam. Contudo, analisar esse volume expressivo de informações mostra-se como um desafio para o professor. Em vista disso, realizou-se um levantamento das mediações pedagógicas que podem ser realizadas junto com a Arquitetura Pedagógica “Debate de Teses”. Para facilitar a aplicação destas mediações, elaborou-se uma proposta de apoio computacional, baseada numa arquitetura de sistemas multiagente, com a finalidade de apoiar o professor na análise das informações produzidas nas interações propostas pela Arquitetura Pedagógica “Debate de Teses”, a fim de propiciar que as mediações pedagógicas identificadas sejam praticadas. A validação da proposta ocorre com a implementação do protótipo computacional de parte da solução, e aplicação deste protótipo a um caso real para coleta de dados. Os dados coletados foram analisados e os resultados obtidos com essa análise apresentam evidências que a solução proposta fornece subsídios para melhorar o processo de mediação pedagógica junto a Arquitetura Pedagógica “Debate de Teses”.

Palavras-chaves. Mediação Pedagógica em Ambientes Virtuais. Arquiteturas Pedagógicas. Arquitetura Pedagógica “Debate de Teses”. Sistemas Multiagente. Recuperação de Informações em Textos.

ABSTRACT

The advent of digital culture has brought new possibilities of interaction and communication between users, promoting the increase of these exchanges and also the increase of the data produced by them. Following this new perspective, interactions in virtual environments for teaching and learning are recorded to enable the teacher to (re)visit the areas and productions of each student seeking information to help them in various ways. Architectures Pedagogic present as a possibility of organizing these interactions, especially Pedagogical Architecture "Debate of Theses" which proposes a flow of interactions, performed by producing texts, to support the construction of knowledge of those who participate. However, analyzing this significant volume of information shows up as a challenge for the teacher. In view of this, we carried out a survey of educational mediations that can be performed along with the Pedagogical Architecture "Debate of Theses". To facilitate the implementation of these mediations, we elaborated a proposal for computer support, based on a multi-agent system architecture, in order to support the teacher in the analysis of the information produced in the proposed interactions for Pedagogical Architecture "Debate of Theses" in order to provide that the identified pedagogical mediations are practiced. The validation of the proposal is to implement the computational prototype of the solution, and application of this prototype to a real case for data collection. The collected data were analyzed and the results obtained from this analysis provide evidence that the proposed solution provides grants to improve the process of pedagogical mediation with the Pedagogical Architecture "Debate of Theses".

Keywords. Pedagogical Mediation in Virtual Environments. Architectures Pedagogic. Pedagogical Architecture "Debate of Theses". Multi-agent Systems. Recovery Information in Texts.

LISTA DE FIGURAS

Figura 1 - Visão geral da Arquitetura Pedagógica “Debate de Teses”.....	18
Figura 2 - Visualização de um Debate finalizado.....	21
Figura 3 - Arquitetura geral da solução proposta por Fernandes Junior.....	22
Figura 4 - Estrutura típica de um Sistema Multiagente.....	32
Figura 5 - Diagrama de Caso de Uso da Solução.....	58
Figura 6 - Diagrama de Casos de Uso AORML - Sistema Multiagente Alpes.....	59
Figura 7 - DCU AORML Agente Verificador.....	60
Figura 8 - DCU AORML Agente de Evolução.....	62
Figura 9 - DCU AORML Agente Indicador.....	64
Figura 10 - DCU AORML Agente Notificador.....	66
Figura 11 - DCU AORML Agente de Contexto.....	67
Figura 12 - Diagrama de Pacotes - Representação do Núcleo de Recuperação de Informações.....	71
Figura 13 - Diagrama de Pacotes – Processamento Textual.....	72
Figura 14 - Fluxo de atividades da técnica Stemming.....	74
Figura 15 - Fluxo de Atividades do Método de Normalização.....	75
Figura 16 - Arquitetura Cliente-Servidor do Protótipo.....	81
Figura 17 - Diagrama de Casos de Uso do Protótipo.....	82
Figura 18: Diagrama de Classes do Protótipo.....	83
Figura 19 - DER Banco de dados ALPES.....	84
Figura 20 - Recorte do DC Protótipo - Classes responsáveis pela limpeza do texto.....	85
Figura 21 - Divisão da amostra em três grupos - Grupo1 (Manual) e Grupo1 (Recurso)....	100
Figura 22 - Divisão da amostra em três grupos - Grupo2 (Manual) e Grupo3 (Recurso)....	101
Figura 23 - Divisão da amostra em três grupos - Grupo 3 (Manual) e Grupo 2 (Recurso). .	101
Figura 24 - Divisão da amostra em três grupos - Grupo1 (Manual) e Grupo3 (Recurso)....	102
Figura 25 - Divisão da amostra em quatro grupos - Grupo1 (Manual) e Grupo1 (Recurso)	102
Figura 26 - Divisão da amostra em quatro grupos - Grupo2 (Manual) e Grupo3 (Recurso)	103
Figura 27 - Divisão da amostra em quatro grupos - Relação entre Grupo3 (Manual), Grupo1 (Recurso) e Grupo4 (Recurso).....	103
Figura 28 - Divisão da amostra em quatro grupos - Grupo4 (Manual) e Grupo2 (Recurso)	104
Figura 29 - Divisão da amostra em cinco grupos - Grupo1 (Manual) e Grupo1 (Recurso). .	104
Figura 30 - Divisão da amostra em cinco grupos - Grupo2 (Manual) e Grupo2 (Recurso). .	105
Figura 31 - Divisão da amostra em cinco grupos - Grupo3 (Manual) e Grupo3 (Recurso). .	105
Figura 32 - Divisão da amostra em cinco grupos - Grupo4 (Manual) e Grupo2 (Recurso). .	105
Figura 33 - Divisão da amostra em cinco grupos - Relação entre Grupo5 (Manual), Grupo4 (Recurso) e Grupo5 (Recurso).....	106

LISTA DE QUADROS

Quadro 1 - Semelhanças e Diferenças – Trabalhos Correlatos x Proposta.....	55
Quadro 2 - Divisão da amostra em três grupos.....	95
Quadro 3 - Divisão da amostra em quatro grupos.....	96
Quadro 4 - Divisão da amostra em cinco grupos.....	97
Quadro 5 - Divisão da amostra em três grupos com o uso do Recurso Grupos de Similaridade.....	98
Quadro 6 - Divisão da amostra em quatro grupos com o uso do Recurso Grupos de Similaridade.....	99
Quadro 7 - Divisão da amostra em cinco grupos com o uso do Recurso Grupos de Similaridade.....	99

LISTA DE SIGLAS

AORML – Linguagem de Modelagem Agente-Objeto-Relacionamento
AP – Arquitetura Pedagógica
APDT – Arquitetura Pedagógica “Debate de Teses”
AVEA – Ambientes Virtuais de Ensino e Aprendizagem
CAA – *Computer Assisted Assessment*.
DC – Diagrama de Classes
DCU – Diagrama de Casos de Uso
DE – Métrica Distância Euclidiana
KM – Algoritmo *K-Means*
LSA – Análise Semântica Latente
MT – Mineração de Textos
NRI – Núcleo de Recuperação de Informações
PLN – Processamento de Linguagem Natural
RI – Recuperação de Informações
SC – Métrica Similaridade de Cossenos
SDT – Sistema “Debate de Teses”
SMA – Sistema Multiagente
SVD – Decomposição de Valor Singular
SVM – Modelo Espaço-Vetorial
TF-IDF – *Term Frequency x Inverse Document Frequency*
UML – Linguagem de Modelagem Unificada

SUMÁRIO

1 INTRODUÇÃO	12
1.1 MOTIVAÇÃO.....	13
1.2 JUSTIFICATIVA.....	13
1.3 OBJETIVO GERAL.....	13
1.3.1 Objetivos específicos	14
1.4 HIPÓTESES DE PESQUISA.....	14
1.5 PROCEDIMENTOS METODOLÓGICOS.....	14
1.6 ORGANIZAÇÃO DO TRABALHO.....	15
2 CARACTERIZAÇÃO DO PROBLEMA	16
2.1 ARQUITETURAS PEDAGÓGICAS.....	16
2.2 ARQUITETURA PEDAGÓGICA “DEBATE DE TESES”.....	17
2.2.1 Suporte computacional à Arquitetura Pedagógica “Debate de Teses”	20
2.3 MEDIAÇÃO PEDAGÓGICA.....	23
2.4 CONSIDERAÇÕES FINAIS.....	30
3 FUNDAMENTAÇÃO TEÓRICA	31
3.1 AGENTES INTELIGENTES E SISTEMAS MULTIAGENTE.....	31
3.1.1 Metodologias para o desenvolvimento de SMA	33
3.2 RECUPERAÇÃO DE INFORMAÇÕES EM TEXTOS.....	39
3.2.1 Clusterização	44
3.3 CONSIDERAÇÕES FINAIS DO CAPÍTULO.....	46
4 TRABALHOS CORRELATOS	47
4.1 LEVANTAMENTO DO ESTADO DA ARTE.....	47
4.2 AS RELAÇÕES ENTRE OS TRABALHOS CORRELATOS E A PROPOSTA DESTA DISSERTAÇÃO.....	54
4.3 CONSIDERAÇÕES FINAIS DO CAPÍTULO.....	56
5 DESCRIÇÃO DA SOLUÇÃO	57
5.1 ARQUITETURA GERAL DA SOLUÇÃO.....	57
5.1.1 Agentes	59
5.1.1.1 Agente Verificador.....	60
5.1.1.2 Agente de Evolução.....	61
5.1.1.3 Agente Indicador.....	64
5.1.1.4 Agente Notificador.....	66
5.1.1.5 Agente de Contexto.....	67
5.1.2 Recursos	68
5.1.2.1 Grupos de Similaridade.....	68
5.1.2.2 Identificação de “similaridades” entre textos.....	69
5.1.2.3 Indicação de Revisores.....	69
5.1.2.4 Análise de Cordialidade.....	70
5.1.2.5 Identificação de Polaridade.....	70
5.1.2.6 Síntese das Autorias.....	70
5.2 NÚCLEO DE RECUPERAÇÃO DE INFORMAÇÕES.....	71
5.2.1 Processamento textual	72
5.2.2 Modelos de Representação	76
5.2.2.1 Módulo LSA.....	76
5.2.2.2 Modelo Espaço-Vetorial – SVM.....	77
5.2.3 Análise de Similaridade	77
5.2.4 Clusterização com K-Means	78

5.3 CONSIDERAÇÕES FINAIS DO CAPÍTULO.....	79
6 PROVA DE CONCEITO.....	80
6.1 PROTÓTIPO COMPUTACIONAL.....	80
6.1.1 Classes responsáveis pela limpeza do texto.....	84
6.1.2 Técnicas de pré-processamento do texto.....	86
6.1.2.1 Etiquetador Morfossintático.....	86
6.1.2.2 Stemming.....	87
6.1.2.3 Normalização.....	87
6.1.3 Análise de Similaridade.....	88
6.1.4 Modelos de Representação.....	88
6.1.4.1 Modelo LSA.....	89
6.1.4.2 Modelo SVM.....	89
6.1.5 Clusterização.....	90
6.2 RECURSOS DO PROTÓTIPO.....	91
6.2.1 Recurso Grupos de Similaridade.....	91
6.2.2 Recurso Síntese das Autorias.....	91
6.3 CONSIDERAÇÕES FINAIS DO CAPÍTULO.....	92
7 APLICAÇÃO DA PROVA DE CONCEITO A UM “DEBATE DE TESES”.....	93
7.1 EXPERIMENTO DE APLICAÇÃO.....	94
7.1.1 Fase 1: Divisão manual dos grupos.....	94
7.1.2 Fase 2: Uso do Recurso Grupos de Similaridade.....	97
7.1.3 Fase 3: Análise dos Resultados.....	99
7.2 CONSIDERAÇÕES FINAIS DO CAPÍTULO.....	107
8 CONCLUSÕES.....	108
8.1 PRODUÇÃO CIENTÍFICA.....	109
8.2 TRABALHOS FUTUROS.....	110
REFERÊNCIAS.....	111

1 INTRODUÇÃO

Com o advento da cultura digital, novas possibilidades de interação tornaram-se possíveis. Estas novas possibilidades trazem consigo a oportunidade de um novo papel para o usuário, o de produtor de informações. Essas produções permeiam os diversos ambientes virtuais com os quais convivemos.

Seguindo essa tendência, os ambientes virtuais utilizados na Educação a Distância permitem o registro das interações realizadas entre os alunos em suas diversas formas de comunicação e troca de informações pelo ambiente. Dentro dessa realidade, o professor pode (re)visitar os espaços e produções de cada aluno buscando informações para auxiliá-los de diversas formas.

Essas novas possibilidades de interação entre alunos e professores apoiam o advento de uma nova forma de “fazer a educação”, onde os espaços físicos e tempos definidos das aulas são abandonados, dando espaço às comunicações assíncronas realizadas através de dispositivos de comunicação diversos (celulares, *tablets*, computadores, etc.) de qualquer lugar e em qualquer tempo.

Com o intuito de organizar as interações realizadas através dos ambientes virtuais de apoio ao ensino para potencializar a construção de conhecimento dos alunos, Carvalho, Nevado e Menezes (2007) elaboram o conceito de Arquiteturas Pedagógicas. Ao utilizar as arquiteturas pedagógicas os alunos interagem seguindo um fluxo de interações determinado, onde as interações podem ser realizadas através da produção de artefatos. Neste trabalho vamos discutir uma delas, a Arquitetura Pedagógica “Debate de Teses”, que será apresentada com mais detalhes no Capítulo 2. Apresenta-se como desafio para o professor monitorar e analisar o grande volume de textos gerados durante as interações propostas pelas arquiteturas pedagógicas.

Faz-se necessário a criação de suporte computacional para facilitar a recuperação das informações contidas nestes textos. Textos, que se tratados corretamente, podem fornecer informações relevantes sobre os alunos como suas necessidades, saberes e interesses, a fim de subsidiar que o professor possa definir novas estratégias para melhorar o processo de ensino e aprendizagem.

1.1 MOTIVAÇÃO

As interações orientadas pela Arquitetura Pedagógica “Debate de Teses” proporcionam uma ampla discussão sobre temas estabelecidos através de afirmações sobre um domínio do conhecimento. Na dinâmica proposta por esta arquitetura pedagógica, as interações entre os alunos são formalizadas com a produção de textos. A tarefa do professor é monitorar o grande volume de textos que são produzidos durante as interações, o que dificulta a prática de mediações pedagógicas. Diante deste cenário, este trabalho pretende colaborar com o Professor em suas atividades de monitoramento e análise das interações, propondo a criação de suporte computacional específico para estas atividades, a fim de facilitar a análise do grande volume de textos que precisam ser lidos e organizados para que o professor possa compreender como os alunos estão construindo seus conhecimentos e auxiliá-los sempre que necessário.

1.2 JUSTIFICATIVA

Através da concepção de suporte computacional que facilite o processo de análise e monitoramento das interações orientadas pela Arquitetura Pedagógica “Debate de Teses”, temos por intenção minimizar os esforços e o tempo gasto pelo professor na prática dessas atividades, possibilitando dessa forma que o professor empregue seu tempo e esforço na melhoria de mediações pedagógicas desenvolvidas em conjunto

1.3 OBJETIVO GERAL

Demonstrar como o uso do suporte computacional, concebido para monitorar e analisar as interações realizadas em ambientes virtuais de ensino e aprendizagem, pode minimizar o esforço e o tempo dedicado pelo professor nestas atividades, e assim favorecer a aplicação de mediações pedagógicas para melhorar o processo de apoio ao ensino e a produção de conhecimento.

1.3.1 Objetivos específicos

- Identificar as mediações pedagógicas que podem ser aplicadas em parceria às arquiteturas pedagógicas;
- Verificar as atividades intrínsecas a cada mediação pedagógica;
- Elaborar uma solução computacional para apoiar a prática das mediações pedagógicas identificadas.

1.4 HIPÓTESES DE PESQUISA

A partir do objetivo geral e dos objetivos específicos, consideramos as seguintes hipóteses para realização desta pesquisa:

- O uso de suporte computacional para monitorar e analisar as interações realizadas pelos alunos em ambientes virtuais mostra-se como uma boa solução para apoiar as atividades do professor neste tipo de ambiente;
- A utilização de suporte computacional viabiliza que o professor desenvolva mediações pedagógicas;
- O processamento de textos baseado na sintaxe é suficiente para destacar semelhanças entre textos e com isso fornecer informações que facilitem o emprego de mediações pedagógicas em ambientes virtuais.

1.5 PROCEDIMENTOS METODOLÓGICOS

Este trabalho se caracteriza como uma pesquisa exploratória, composta pela elaboração de uma solução computacional, a aplicação desta solução a um caso real, bem como a coleta e análise dos resultados obtidos com o experimento.

Realizamos um estudo sobre a utilização da Arquitetura Pedagógica “Debate de Teses” através de um sistema computacional desenvolvido para simulá-la. Neste estudo buscamos identificar quais mediações pedagógicas poderiam ser aplicadas em paralelo à arquitetura pedagógica.

Em sequência, elaboramos um modelo computacional para apoiar a prática dessas mediações, e implementamos um protótipo funcional do mesmo utilizando tecnologias livres.

Por fim, realizamos um experimento para comprovar a viabilidade de nosso objetivo de pesquisa.

1.6 ORGANIZAÇÃO DO TRABALHO

Este trabalho está organizado da seguinte forma: O Capítulo 2 apresenta a caracterização do problema, descrevendo a parte inicial desta pesquisa.

No Capítulo 3 apresentamos o referencial teórico computacional utilizado como base para a construção da solução. Evidenciamos, no Capítulo 4, os trabalhos correlatos a esta proposta, com destaque para as principais semelhanças e diferenças entre os trabalhos selecionados e a presente pesquisa.

Em sequência, no Capítulo 5 descrevemos com minúcias a solução elaborada. No Capítulo 6 detalhamos o desenvolvimento do protótipo computacional da solução. E no Capítulo 7 apresentamos a condução do experimento de aplicação do protótipo a um caso real para realizarmos a coleta e posterior análise dos dados obtidos.

Por fim, no Capítulo 8 estão as conclusões dessa pesquisa, ressaltando as hipóteses confirmadas ou descartadas, os objetivos alcançados, as limitações do projeto, bem como as produções científicas frutos desta pesquisa e os trabalhos futuros.

2 CARACTERIZAÇÃO DO PROBLEMA

Apresentamos neste capítulo a descrição do contexto do problema de pesquisa identificado, primordiais para fundamentar o objetivo geral e justificativa de pesquisas apresentadas no Capítulo 1. Na Seção 2.1 apresentamos os conceitos sobre as Arquiteturas Pedagógicas. Na Seção 2.2 descrevemos e detalhamos o cerne do contexto de pesquisa, a Arquitetura Pedagógica “Debate de Teses”. Através da descrição das Mediações Pedagógicas, na Seção 2.3 evidenciamos a parte inicial desta pesquisa. Por fim, na Seção 2.4 fazemos as considerações finais do capítulo.

2.1 ARQUITETURAS PEDAGÓGICAS

As diversas formas de promover os processos de ensino e aprendizagem através do uso de tecnologias digitais da informação e comunicação, mais especificamente pelo uso de computadores, caracterizam-se pela facilidade de acesso e aplicação, por permitir acesso de múltiplos usuários, por possuir ferramentas de autoria e por fornecer suporte para construções coletivas, ou seja, por apoiar a elaboração de ideias desenvolvidas de forma coletiva entre participantes de uma determinada atividade pedagógica. Assim sendo, estes ambientes podem ser vistos como sistemas colaborativos que auxiliam na construção de conhecimento e que potencializam o emprego de teorias de aprendizagem baseadas na interação.

Considerando o cenário da Educação a Distância que utiliza como aporte teórico para a aprendizagem teorias construtivistas baseadas na Epistemologia Genética de Piaget (apud RAMOZZI-CHIAROTTINO, 1988), e que se caracterizam pelo uso de tecnologias da informação e comunicação para sua aplicação, Carvalho, Nevado e Menezes (2007) apresentam o conceito de Arquiteturas Pedagógicas (AP).

As arquiteturas pedagógicas são, antes de tudo, estruturas de aprendizagem realizadas a partir de confluências de diferentes componentes: **abordagem pedagógica, software educacional, internet, inteligência artificial, Educação a Distância, concepção de tempo e espaço**. O caráter destas arquiteturas pedagógicas é pensar a aprendizagem como um trabalho artesanal, construído na vivência de experiências e na demanda de ação, interação e meta-reflexão do sujeito sobre os fatos, os objetos e o meio ambiente sócio-ecológico (CARVALHO; NEVADO; MENEZES, 2007, p. 39, grifo nosso).

Dessa forma, as AP apresentam-se como alternativas para a organização das interações realizadas em Ambientes Virtuais de Ensino e Aprendizagem (AVEA), a partir de um fluxo de interações pré-determinadas e que tenham por objetivo apoiar a construção de conhecimento dos indivíduos que dela participam, de forma colaborativa e autônoma, a partir das suas interações com o meio virtual, com outros indivíduos e com materiais diversos.

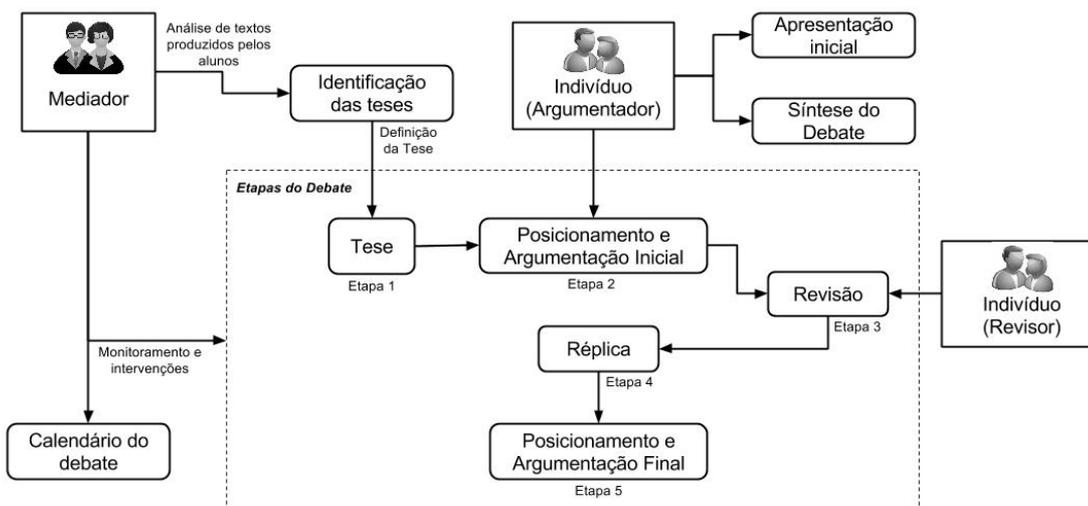
2.2 ARQUITETURA PEDAGÓGICA “DEBATE DE TESES”

Elaborada de acordo com a Teoria Piagetiana (PIAGET, apud RAMOZZI-CHIAROTTINO, 1988), na qual entende-se que a construção de conhecimento requer que os sujeitos realizem interações que provoquem desequilíbrios em seus conhecimentos prévios, e que diante disso o indivíduo busque as informações necessárias para entrar em equilíbrio novamente, consolidando assim seu conhecimento sobre um determinado assunto. A Arquitetura Pedagógica “Debate de Teses” (APDT) “[...] busca apoiar as aprendizagens, sistematizando interações que favoreçam/incentivem esses movimentos num processo de construção conjunta do conhecimento” (NEVADO; MENEZES; VIEIRA JUNIOR, 2011, página 829).

A APDT foi concebida, no contexto da cultura digital, onde os sujeitos interagem de forma assíncrona e geograficamente distribuídos com a mediação de redes de comunicação e propõe a construção de conhecimento dos participantes de forma colaborativa através de suas interações orientadas. Seus elementos estruturantes foram pensados de forma que o indivíduo contribua efetivamente com a construção de seu conhecimento.

O foco principal é que o indivíduo, partindo de seu conhecimento prévio, estenda e aprofunde esses conhecimentos através das interações com outros indivíduos, seguindo uma dinâmica preestabelecida. Nestas interações, que são feitas através da produção de textos, os indivíduos expõem suas convicções a respeito de teses (afirmações) propostas pelo mediador (professor). Uma visão geral sobre a APDT com suas interações definidas como etapas, é apresentada pela Figura 1.

Figura 1 - Visão geral da Arquitetura Pedagógica “Debate de Teses”



Fonte: Autoria Própria

O Mediador (professor ou responsável pelo debate), é o responsável por analisar os textos produzidos pelos indivíduos (alunos e demais participantes do debate), e assim identificar uma afirmação que se destaque e que, de acordo com o entendimento do mesmo, possibilite gerar dúvidas e incertezas nos indivíduos participantes do debate. Neste entendimento, uma Tese é uma afirmação sobre qualquer assunto, sobre a qual se pode concordar ou discordar. O mediador, em comum acordo com os participantes, define as Teses que serão debatidas – **Etapa 1** - e iniciam-se as interações realizadas pelos indivíduos. Após a Etapa 1, o mediador monitora e realiza mediações (intervenções) pedagógicas individuais ou coletivas, considerando os pressupostos da APDT, até a sua finalização.

O Indivíduo, antes de iniciar sua participação no debate, elabora um texto com as suas concepções iniciais gerais em relação ao tema do debate (Apresentação Inicial). A partir desse momento, na **Etapa 2 – Posicionamento e Argumentação Inicial** – cada indivíduo, no papel de Argumentador, informa se concorda (“a favor”) ou discorda (“contra”) da tese proposta, e descreve de forma lógica seus argumentos, com base em seus conhecimentos prévios, para fundamentar seu posicionamento.

Na Revisão – **Etapa 3** – o indivíduo deve analisar a consistência dos argumentos de outros indivíduos, geralmente dois, tendo por base verificar se há evidências que sustentem as argumentações apresentadas. Os participantes na etapa de revisão

assume o papel de Revisor. Essa etapa caracteriza as primeiras interações para construção de conhecimento de cada indivíduo, uma vez que ao ter acesso as argumentações dos seus revisados, o indivíduo depara-se com outras opiniões, ou pontos de vistas diferenciados, ou explicações distintas sobre um mesmo ponto de vista, ou mesmo opiniões contrárias a sua e que foram fundamentadas de forma lógica e coerente. Isto pode contribuir para que o indivíduo coloque em dúvida o seu próprio posicionamento.

Na **Etapa 4** – Réplica - o indivíduo, de volta ao seu papel de Argumentador, deve responder aos questionamento e/ou críticas feitos por seus revisores. Nesta etapa, o aluno tem acesso a novos pontos de vista e argumentos sobre a tese que foram tecidos por seus revisores e que são diretamente relacionados à sua argumentação inicial.

Com esta sistemática, um indivíduo tem acesso a quatro produções realizadas por seus colegas de debate, duas no momento em que revisa as argumentações de seus pares, outras duas quando analisa as revisões feitas sobre a sua argumentação. Estas duas Etapas, Etapa 3 e Etapa 4, formalizam a troca de informações e interações entre os indivíduos durante um debate.

Ao fim da Etapa 4, o indivíduo deve estar preparado para a reescrita de seu posicionamento e argumentação, que é feito na **Etapa 5** – Posicionamento e Argumentação Final. Nesta última etapa, o indivíduo indica novamente seu posicionamento sobre a tese e reescreve sua argumentação, levando em conta as interações realizadas. As interações do debate terminam.

Com o objetivo de consolidar os novos conhecimentos adquiridos pelo indivíduo durante o debate, este é convidado a produzir seu último texto, a Síntese do Debate. Neste texto ele descreve de forma clara e resumida seu processo de aprendizagem e sua evolução em relação ao tema abordado pela tese, destacando pontos que considere importantes.

Há variações de condução da APDT onde pode ser desenvolvido um debate coletivo e/ou podem ser sugeridas leituras complementares ao assunto, antes que se chegue à Etapa 5.

As interações descritas ocorrem mediante a autoria de textos, e para que sejam

aplicadas mediações pedagógicas pertinentes a cada etapa, o mediador deve ler e compreender todos os textos produzidos. Considerando a aplicação da APDT numa turma com 10 (dez) indivíduos e seguindo as Etapas do Debate (Figura 1), para cada tese proposta o professor (mediador) terá: 10 (dez) posicionamentos e argumentações iniciais, 20 (vinte) revisões, 20 (vinte) réplicas, 10 (dez) posicionamentos e argumentações finais. Um total de 60 (sessenta) textos para analisar, compreender e identificar as características intrínsecas a cada etapa.

A prática da APDT é desejável, contudo sua operacionalização através de ferramentas computacionais como *wikis* e fóruns torna-se complexa e custosa para o mediador. À vista disso, duas soluções computacionais foram criadas para facilitar sua aplicação (NEVADO; MENEZES; VIEIRA JUNIOR, 2011; FERNANDES JUNIOR; MENEZES, 2015; FERNANDES JUNIOR, 2015). A seguir descremos tais suportes computacionais.

2.2.1 Suporte computacional à Arquitetura Pedagógica “Debate de Teses”

Em Nevado, Menezes e Vieira Junior (2011), além do conceito da Arquitetura Pedagógica Debate de Teses é apresentado um ambiente na *web* desenvolvido para dar suporte à sua aplicação. Este ambiente foi concebido com base na arquitetura cliente-servidor, e possui três subambientes: Administrador, Moderador e Participante (NEVADO; MENEZES; VIEIRA JUNIOR, 2011). Iremos nos referir a este protótipo como Sistema “Debate de Teses”¹ (SDT).

Dentro do SDT todas as funcionalidades necessárias para criação de debates e cadastramento de usuários foram implementadas. Mais detalhes podem ser verificados em (NEVADO; MENEZES; VIEIRA JUNIOR, 2011). Diante disso, algumas restrições foram aplicadas, como a definição padrão de dois revisores para cada argumentador. Aspectos de coordenação e flexibilidade não são explorados no SDT.

Na Figura 2, apresentamos a parte de um debate feito com o SDT, onde conta a Tese proposta, o Posicionamento Inicial, a Argumentação Inicial, as Revisões, as

1 A versão atualizada do protótipo pode ser acessada em <http://www.pead.faced.ufrgs.br/sites/cms/debate/> ou em <http://lied.inf.ufes.br/debate2/>. Acessos em 17 de fevereiro de 2016.

Réplicas e Posicionamento Final.

Figura 2 - Visualização de um Debate finalizado

Página Marcelo Novaes

Argumentador: Marcelo Novaes
Mediador: Credine Menezes
Revisores: As identidades dos revisores estão ocultas

Teses	Posicionamento Inicial	Argumento	Revisão	Réplica	Posicionamento Final
A aprendizagem cooperativa, além de promover a construção de conhecimento baseado na diversidade de idéias, ajuda no processo de socialização dos participantes.	Concordo	Acredito que a socialização seja algo adquirido, e as atividades realizadas no aprendizado colaborativo promovem este exercício.	Não acha que a cooperação vai além da socialização? As atividades cooperativas podem ajudar a formular novas ideias e até mesmo mudar velhos conceitos? Como acha que isso pode ser conseguido com atividades de aprendizagem cooperativa?	A cooperação vai além da socialização porém, se bem mediada e bem planejadas, as atividades cooperativas promovem, necessariamente, algum nível de socialização.	Concordo. Apesar de a socialização não ser o único ponto relacionado com a aprendizagem colaborativa, ela é um dos pontos centrais. Sempre é possível que não haja uma socialização efetiva em um cenário no qual os participantes não a querem, porém as dinâmicas colaborativas, se bem planejadas, fazem com que a socialização acabe ocorrendo, em algum nível pelo menos.
			Concordo com o posicionamento do argumentador e saliento que tanto a aprendizagem cooperativa quanto socialização dos participantes envolvidos depende também do engajamento dos participantes da atividade em questão.	O ponto é exatamente este, as atividades devem ser bem planejadas e mediadas para que haja este engajamento.	

Fonte: SISTEMA DEBATE DE TESES, 2016

No SDT a Fase de Posicionamento e Argumentação Final foi dividida para facilitar a leitura do mediador. E a fase Posicionamento e Argumentação Final é representada apenas como Posicionamento Final.

Encontramos outra implementação computacional da APDT em Fernandes Junior e Menezes (2015). Essa nova implementação tem como principal foco a coordenação e flexibilidade das atividades do debate. Os autores destacam que “[...] o esforço de coordenação não é trivial para um professor sem auxílio de um software para mediar os envolvidos [...]” (FERNANDES JUNIOR; MENEZES, 2015, p. 703). Diante disso, a pesquisa explora formas de auxiliar na coordenação de atividade como a escolha da tese que será debatida e o emparceiramento de revisores, com maior enfoque para a segunda.

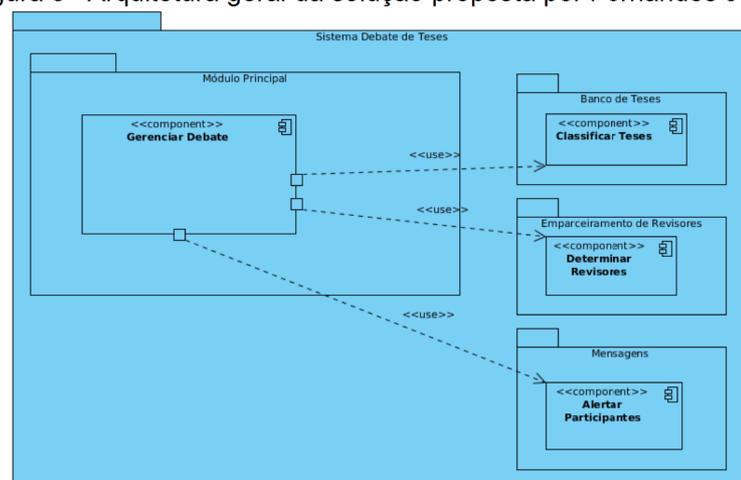
Fernandes Junior e Menezes (2015) propõem quatro formas de realizar o emparceiramento de revisores, sendo:

1. Distribuição Aleatória: Não são considerados critérios para realizar a distribuição, apenas é considerado que todos a todos os alunos sejam atribuídos revisores.
2. Distribuição Aleatória evitando pares idênticos: Igual à Distribuição Aleatória acrescida de uma regra para evitar que o participante A seja definido como revisor do participante B, e o participante B seja definido como revisor do participante A.

3. Distribuição que evita a formação de subgrupos de revisores: Segue as distribuições anteriores acrescentando como nova regra a verificação de formação de subgrupos. Como exemplo, temos os participantes A, B e C, relacionados como: A revisa B e C, B revisa A e C, C revisa B e A. Essa distribuição tem como premissa evitar que isso aconteça.
4. Distribuição com base na qualidade de argumentação: A distribuição é realizada a partir de uma nota, entre zero e dez, atribuída pelo professor a cada argumentação inicial produzida pelos alunos. Sendo possível definir, com base nas notas, os parâmetros para a distribuição.

À vista disso, Fernandes Junior (2015) apresenta a arquitetura geral da solução proposta (Figura 3). A solução computacional é composta por quatro módulos: Principal, Banco de Teses, Emparceiramento de Revisores e Mensagens. Os módulos são compostos por funcionalidades que garantem a configuração das teses para posterior seleção, configuração do debate, cadastramento de participantes, troca de mensagens entre professor e participantes e opções para definição de revisores. Também destacamos a flexibilidade na definição de fases de interação, uma vez que a solução possibilita a configuração de várias fases de réplica (réplica, tréplica, ..., néplica) (FERNANDES JUNIOR, 2015).

Figura 3 - Arquitetura geral da solução proposta por Fernandes Junior



Fonte: FERNANDES JUNIOR, 2015.

As duas soluções computacionais apresentadas facilitam a aplicação da APDT organizando as interações e armazenando-as, possibilitando que o professor acesse

as produções realizadas a qualquer tempo. Essa garantia de acesso viabiliza que novas orientações e novas abordagens sobre um tema já discutido em um debate sejam trabalhadas com os alunos. Assim, o professor poderá aprofundar as discussões e tratar tópicos inexplorados.

Além disso, a utilização da APDT remete ao professor a possibilidade de realizar mediações pedagógicas pontuais ou coletivas junto aos participantes, ao final ou durante a realização de cada uma das etapas do debate. Contudo, a prática dessas mediações são custosas em relação a tempo e quantidade de ações que devem ser realizadas para sua aplicação.

2.3 MEDIAÇÃO PEDAGÓGICA

Durante o estudo realizado sobre a Arquitetura Pedagógica “Debate de Teses”, identificamos as necessidades e/ou possibilidades para a realização de mediações pedagógicas. Diante disso, nesta seção apresentamos o levantamento das situações que podem gerar mediações pedagógicas. As mediações listadas têm como objetivos: analisar, compreender e fornecer *feedback* para os indivíduos, e são realizadas pelo professor (mediador) visando auxiliar no processo de construção de conhecimento dos indivíduos que participam da APDT.

Segundo Pérez e Castilho (1999),

“A mediação pedagógica busca abrir caminho a novas relações do estudante com os materiais, com o próprio contexto, com outros textos, com seus companheiros de aprendizagem, incluindo o professor, consigo mesmo e com seu futuro” (PÉREZ; CASTILHO, 1999, p. 10).

De forma geral, as mediações pedagógicas podem ser vistas como intervenções individuais ou coletivas. Contudo, de acordo com os preceitos das arquiteturas pedagógicas, o professor deve se ater a realizar intervenções que ajudem os indivíduos na construção de seu conhecimento e não a limitá-la.

Listam-se abaixo as mediações pedagógicas elicitadas, com suas respectivas descrições e cenários de aplicação.

1. Mediação: Analisar autorias feitas nas etapas

Descrição: Verificar se os textos produzidos pelos indivíduos contém os elementos característicos da etapa do debate a que se referem.

Cenário: Na argumentação inicial um indivíduo deve se posicionar “a favor” ou “contra” a tese e fundamentar seu posicionamento, apresentando de forma lógica seus argumentos, baseado em seus conhecimentos prévios sobre o assunto. Neste caso, o professor poderá analisar se os textos produzidos pelos alunos estão relacionados com o assunto da tese, se possuem elementos que apóiam o posicionamento escolhido, e apontar para um grupo de alunos ou mesmo para um indivíduo quais elementos não foram relacionados e descritos a fim de auxiliar a construção de uma argumentação mais completa.

2. Mediação: Identificar tópico para orientar a Síntese do Debate

Descrição: Ao analisar as produções realizadas na Síntese do Debate, o professor identifica tópicos sobre o tema da tese que não foram abordados pelo indivíduo.

Cenário: Após a conclusão do debate, o indivíduo produz uma pequena resenha sobre seu aprendizado. O mediador, ao identificar que o indivíduo deixou de escrever sobre determinado tópico sobre o assunto abordado pela tese, aponta quais foram os tópicos. Assim, o indivíduo deverá considerar esses tópicos durante a elaboração de sua síntese.

3. Mediação: Verificar a evolução das argumentações

Descrição: O professor identifica quais elementos apresentados na argumentação inicial foram melhorados, contrapostos ou deixaram de fazer parte da argumentação final. E se o posicionamento inicial em relação à tese mudou.

Cenário: A partir da compreensão do professor sobre a argumentação inicial de um indivíduo, o mesmo destaca os elementos expostos e os compara com os elementos contidos na argumentação final. Assim, destaca quais elementos foram melhor descritos, quais foram contrapostos e quais não

fazem parte da argumentação final, pretendendo traçar uma linha evolucionária sobre a argumentação do indivíduo. Quando novos elementos forem identificados, é necessário analisar as revisões e réplicas, para verificar a origem daquele elemento e de que forma as revisões e réplicas contribuíram para essa evolução.

4. Mediação: Informar o prazo das etapas e monitorar os envios

Descrição: O professor envia lembretes para os indivíduos que ainda não tiverem realizado sua autoria na etapa quando faltar 1 (um) dia para o encerramento dela.

Cenário: Ao definir o calendário do debate, o professor determina as datas máximas para elaboração de cada etapa. Quando um indivíduo envia a produção de uma etapa, o professor é avisado e assim monitora quantos indivíduos ainda não enviaram aquela etapa. Dessa forma, ele pode alertar, através do ambiente ou por outro meio de comunicação (ex.: e-mail), àqueles indivíduos que ainda não fizeram suas produções referentes à etapa que está ativa.

5. Mediação: Elaborar síntese das argumentações finais para realizar o *feedback* coletivo

Descrição: Identificar e destacar os elementos semelhantes entre as argumentações finais enviadas pelos indivíduos. Agrupar, com base na semelhança, as argumentações finais.

Cenário: A partir da compreensão do professor sobre as argumentações finais dos indivíduos, este destaca os elementos semelhantes entre as argumentações e agrupa as argumentações finais mais semelhantes. Dessa forma, o professor pode elaborar um *feedback* coletivo sobre aquelas argumentações e enviar para os indivíduos, reduzindo seu esforço de elaboração e envio de *feedback* individualizado.

6. Mediação: Agrupar indivíduos por posicionamento e argumentação inicial e distribuir revisores

Descrição: O professor agrupa os indivíduos de acordo com seu

posicionamento “a favor” ou “contra” a tese proposta.

Cenário: O indivíduo posiciona-se “a favor” ou “contra” o tema proposto pela tese e fundamenta seu posicionamento através de sua argumentação. A partir disso, o professor separa os indivíduos de acordo com o posicionamento indicado, e distribui como revisores para um indivíduo, dois outros indivíduos que tenham posicionamento contrário ao seu. Dessa forma, o professor utiliza os posicionamentos e argumentações para colocar em prática os princípios Piagetianos de “desequilíbrios”, indicando revisores que não estão num mesmo grupo a fim de estimular o acesso a pontos de vistas diferentes sobre uma mesma tese. Assim, os revisores irão contrapor os argumentos do indivíduo revisado. Por consequente, na réplica, o indivíduo revisado terá acesso a contrapontos sobre seu posicionamento e argumentação, o que causará certo desequilíbrio. Dessa forma, o indivíduo revisado terá que buscar por novos conhecimentos para conseguir elaborar suas réplicas, estabelecendo novamente um estado de equilíbrio sobre o assunto.

7. Mediação: Agrupar indivíduos que possuem argumentações semelhantes

Descrição: A partir da análise da argumentação inicial ou final, o professor destaca os principais elementos que fundamentam o posicionamento indicado.

Cenário: Ao analisar a argumentação inicial de um indivíduo, o professor destaca os principais fundamentos apresentados que embasam o posicionamento indicado. E, para indivíduos que tenham o mesmo posicionamento, “a favor” ou “contra”, o professor relaciona as fundamentações semelhantes realizadas. Ou seja, o indivíduo “A”, que se posicionou “a favor” da tese, elaborou sua argumentação com elementos semelhantes ao do indivíduo “B” que também se posicionou “a favor” da tese. Agrupando esses indivíduos, o professor identifica aqueles que deram início ao debate com nível de conhecimento semelhante. E, ao analisar as argumentações finais, o professor destaca os mesmos elementos, e verifica se os indivíduos agrupados com base na argumentação e posicionamento

inicial, continuam agrupados ao finalizar o debate, ou seja, se o posicionamento e argumentação final são também semelhantes.

8. Mediação: Fazer comentários sobre a argumentação inicial para orientar o indivíduo

Descrição: O professor analisa as argumentações iniciais dos indivíduos com o objetivo de identificar quais indivíduos elaboraram argumentações destoantes da tese, ou que descreveram de forma sucinta seu entendimento sobre o tema.

Cenário: A partir da análise e compreensão do professor sobre a argumentação inicial do indivíduo, o professor relaciona o contexto descrito na argumentação com o contexto da tese proposta, e verifica se o indivíduo elaborou uma argumentação coerente com o assunto da tese. Se a argumentação não for coerente com a tese, o professor informa o indivíduo que sua argumentação não está de acordo e pede que ele a refaça. Além disso, o professor pode indicar para aqueles que tenham elaborado uma argumentação sucinta, elementos em que podem melhorá-la ou provocar o indivíduo a desenvolver a argumentação com outros elementos ampliando a sua explicação.

9. Mediação: Verificar se as revisões feitas pelo indivíduo são diferentes e relevantes

Descrição: O mediador analisa as revisões feitas por um mesmo indivíduo buscando se ele questiona a argumentação e faz críticas. Além disso verifica se as duas revisões são diferentes e relacionadas à argumentação a que está vinculada.

Cenário: Dentro dessa arquitetura pedagógica, cada indivíduo é responsável por revisar a argumentação de outros dois indivíduos (revisão por pares), dessa forma o indivíduo mantém contato outras duas opiniões sobre a tese. Contudo, a construção da revisão tem como foco questionar os argumentos utilizados pelo indivíduo revisado para fundamentar seu posicionamento. Isto posto, é possível que um indivíduo realize a revisão de outros dois indivíduos que tenham opiniões semelhantes, e queira se poupar do esforço de escrever

outra revisão, por considerá-las muito semelhantes, elaborando duas revisões iguais. O professor analisa as revisões feitas, e caso identifique que as duas revisões são iguais ou muito parecidas, contata o indivíduo revisor e solicita que o mesmo refaça uma das revisões.

10. Mediação: Conferir se as réplicas esclarecem os questionamentos levantados nas revisões

Descrição: O mediador analisa as réplicas feitas pelo indivíduo, correlacionando-as com as revisões às quais estão vinculadas. As réplicas devem esclarecer os questionamentos feitos nas revisões.

Cenário: Ao analisar as réplicas produzidas por um indivíduo, o professor verifica se os questionamentos levantados nas revisões são respondidos de forma esclarecedora e lógica.

11. Mediação: Verificar qual o grau de influência das revisões e réplicas na argumentação final

Descrição: Essa análise é feita para verificar quais elementos inseridos na argumentação final foram apresentados nas fases de revisão e réplica.

Cenário: Ao analisar o posicionamento e argumentação final de um indivíduo, o professor destaca os elementos utilizados para fundamentar seu posicionamento, e no caso de haver elementos novos que não foram relatados na argumentação inicial, o professor verifica se esses elementos aparecem nas etapas de revisão e/ou réplica. Assim pode-se verificar qual o nível de influência das trocas realizadas nessas etapas, e como as trocas auxiliaram a construção da argumentação final do indivíduo. Uma argumentação final que possui muitos elementos distintos da argumentação inicial, indica uma mudança de pensamento sobre o tema, mesmo que o posicionamento permaneça o mesmo. E, se esses novos elementos tiverem sido apresentados nas fases de revisão e réplica, demonstra que essas fases influenciaram a elaboração da argumentação final.

12. Mediação: Identificar quais teses podem ser trabalhadas a partir das argumentações finais

Descrição: Analisar as argumentações finais e destacar novas teses a partir das produções realizadas pelos indivíduos

Cenário: Através da análise dos argumentos finais, o professor identifica quais foram os assuntos que ficaram mais e menos evidentes nas produções, e escolhe quais as próximas teses podem ser trabalhadas sobre este mesmo tema para auxiliar na construção do conhecimento dos indivíduos.

13. Mediação: Verificar se houve cordialidade nas revisões e réplicas

Descrição: Analisar as revisões e réplicas a fim de identificar se contém palavras de baixo calão, termos agressivos ou foram escritas de forma ofensiva.

Cenário: Essa intervenção visa prevenir e alertar os indivíduos caso sejam identificados termos de baixo calão (palavrões), termos agressivos, foram escritas de forma ofensiva ou se há comparações entre as argumentações revisadas que as denigram. Por exemplo, um indivíduo revisará a argumentação inicial de outros dois indivíduos, A e B, na revisão não é apropriado que o indivíduo revisor faça comparações entre as argumentações a que teve acesso, ou seja, diga que a argumentação inicial de A é melhor que a de B, ou vice e versa.

Empregar as mediações pedagógicas de forma manual, ou seja, com o professor acessando as produções de cada indivíduo, lendo, compreendendo e destacando características, demandam um tempo relevante do professor, o que dificulta sua prática, uma vez que o professor desenvolve outras atividades além do acompanhamento desta atividade em específico.

Uma forma de diminuir o tempo dedicado à aplicação das mediações pedagógicas e torná-las uma prática viável é utilizar a tecnologia como apoio para realização dessas tarefas, através da construção de ferramentas computacionais capazes de analisar e identificar características nos textos, que assim minimizem os esforços do professor.

2.4 CONSIDERAÇÕES FINAIS

Com a intenção de contribuir para a viabilidade de aplicação das mediações pedagógicas descritas, junto com a Arquitetura Pedagógica “Debate de Teses”, tomamos como fonte de pesquisa e motivação as atividades propostas por estas mediações para a construção de um suporte computacional, ainda inexistente para esse tipo de ambiente.

Apresentamos a seguir o levantamento teórico utilizado como alicerce para construção de tal solução.

3 FUNDAMENTAÇÃO TEÓRICA

Neste capítulo descrevemos os fundamentos teóricos utilizados para a construção de uma solução computacional inteligente. Logo, na Seção 3.1 trazemos o conceito de Agentes Inteligentes e Sistemas Multiagente e um breve apanhado sobre as metodologias propostas para a criação deste tipo de sistema. Na Seção 3.2 apresentamos um resumo sobre Recuperação de Informações em Textos. E, na Seção 3.3 temos as considerações finais do capítulo.

3.1 AGENTES INTELIGENTES E SISTEMAS MULTIAGENTE

Na literatura, encontramos muitas definições para o termo Agente que têm como característica comum atribuir ao Agente autonomia em suas ações dentro de um sistema. Consideramos a definição apresentada por Wooldridge (2009) para o termo Agente: “Um agente é um sistema de computador que está situado em algum ambiente, e que é capaz de ação autônoma neste ambiente, a fim de cumprir os objetivos para os quais fora projetado [...]” (WOOLDRIDGE, 2009, p. 15, tradução nossa).

Contudo, as referências a Agentes e Agentes Inteligentes se diferem na literatura em relação às características intrínsecas de cada um. Dessa forma, em Wooldridge e Jennings, citado por Wooldridge (2009, p. 26), para considerarmos que um agente possui inteligência, ou seja, um Agente Inteligente, o agente deve possuir junto com a autonomia, as seguintes características:

- **Reatividade:** Os agentes inteligentes são capazes de perceber seu ambiente e responder em tempo hábil às mudanças que ocorrem nele, a fim de satisfazer seus objetivos de projeto.
- **Proatividade:** Os agentes inteligentes são capazes de apresentar um comportamento direcionado por objetivo, tomando a iniciativa, a fim de satisfazer seus objetivos de projeto.
- **Habilidades sociais:** Os agentes inteligentes são capazes de interagir com outros agentes (e possíveis seres humanos), a fim de satisfazer seus

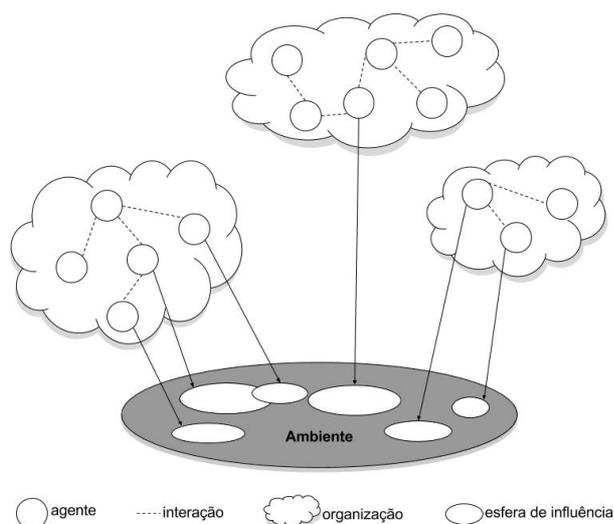
objetivos de projeto.

Agentes inteligentes podem ser classificados em dois tipos: Agentes Inteligentes Reativos e Agentes Inteligentes Cognitivos. Os Agentes Inteligentes Reativos são aqueles que reagem mediante a condição percebida em um cenário real ou fictício. Os Agentes Inteligentes Cognitivos são aqueles que consideram uma série de fatores para realizar uma reação complexa e bem elaborada. Além disso, um agente inteligente deve possuir atributos que o possibilitem interagir com o meio, são eles os atributos sensores e atributos atuadores. Atributos sensores permitem que os agentes inteligentes capturem informações do meio sobre sua influência. Atributos atuadores possibilitam a aplicação das reações dos agentes inteligentes.

Reis (2003) define Sistemas Multiagente (SMA) como “Um Sistema Multiagente é um sistema computacional em que dois ou mais agentes interagem ou trabalham em conjunto de forma a desempenhar determinadas tarefas ou satisfazer um conjunto de objetivos [...]” (REIS, 2003, p. 50).

Os agentes inteligentes que compõem um SMA possuem uma área de atuação ou influência, que pode ser compartilhada com outros agentes inteligentes, e que fazem parte do SMA como um todo, como mostra a Figura 4, um redesenho da Estrutura Típica de um SMA proposto por Jennings (2000).

Figura 4 - Estrutura típica de um Sistema Multiagente



Fonte: Adaptado de JENNINGS, 2000, p. 281

A interação entre agentes é um processo multidimensional, que contém em seu nível mais básico a comunicação. A comunicação pode ser classificada como indireta, quando um agente altera o ambiente e outro agente capta a alteração, ou direta, quando é realizada por meio da troca de mensagens entre dois ou mais agentes (MAGALHÃES NETTO, 2006).

Segundo Coppin (2004) em muitas situações, um SMA pode ser composto apenas por agentes reativos, uma vez os agentes não existem de forma isolada e na prática, os agentes são compostos por um conjunto de regras, logo suas ações são limitadas. Ainda assim, SMA são considerados sistemas complexos, tanto para sua modelagem, quanto para seu desenvolvimento. Para isso, metodologias que apoiam a elaboração de projeto de *software* para SMA foram desenvolvidas, conforme apresentamos a seguir.

3.1.1 Metodologias para o desenvolvimento de SMA

Segundo Castro, Alencar e Silva (2006) a necessidade de criação de um novo paradigma para orientar o desenvolvimento de *software* foi necessária devido às complexidades inerentes a um Sistema Multiagente. Dessa forma, o paradigma orientado a agentes é considerado uma evolução dos paradigmas anteriores e tem por objetivo nortear a representação da análise, projetos e construção de sistemas de software complexos.

Partindo dos modelos e notações da UML (*Unified Modeling Language* – Linguagem de Modelagem Unificada), várias linguagens e metodologias foram propostas para serem aplicadas a projetos de SMA. A seguir apresentamos as principais linguagens para definição de SMA baseadas na UML, com destaques para suas características. E em seguida, apresentamos as principais Metodologias de Engenharia de Software Orientada a Agentes. (GUEDES, 2012).

- **Linguagens derivadas da UML**

- *AUML – Agent UML*

- Proposta por Huget e Odell (apud Guedes, 2012). Não possui uma notação formal e documentada. Utiliza da UML o Diagrama de Classes e

os esteriótipos de classes e objetos para representar agentes. Usa o Diagramas de Máquinas de Estado para modelar o comportamento e Diagramas de Interação para modelar a comunicação entre os agentes. Não suporta abstração para modelagem de agentes cognitivos. A AUML está atualmente inativa.

- *AORML – Agent-Object-Relationship Modeling Language – Linguagem de Modelagem Agente-Objeto-Relacionamento*

Proposta por Wagner e Taveter (apud Guedes, 2012), tem por objetivo suportar o projeto de alto nível de SMA. Baseia-se sobre o metamodelo AOR – *Agent-Object-Relationship*, cujo propósito é fornecer uma metodologia genérica para análise e projeto de SMA. Possui notação própria e oferece sete diagramas para criação de seus modelos, divididos em dois grupos, Modelos Internos e Modelos Externos. Os diagramas de Modelos Externos especificam um agente ou um grupo de agentes para o qual se deseja desenvolver um modelo de estado e comportamento. Os diagramas de Modelos Internos descrevem o mundo como ele pode ser representado pelo estado mental do agente em foco. O Diagrama de Caso de Uso da UML é utilizado sem nenhum tipo de adaptação, nele os agentes externos ao sistema são representados como atores, os casos de uso representam as funcionalidades oferecidas pelo sistema e as interações entre os agentes. Os agentes são representados pela fronteira do sistema.

- *AML – Agent Modeling Language*

Proposta por Cervenka e Trencansky (apud Guedes, 2012) é uma linguagem visual semi-formal para especificar, modelar e documentar sistemas que possuem características da teoria de SMA. É especificada como uma extensão à UML 2.0, e utiliza adaptações do Diagrama de Classes para descrever os diversos aspectos de um SMA, adaptações do Diagrama de Sequência para descrever as interações entre os agentes, e adaptações do Diagrama de Atividades. Além disso faz uso de notações UML adaptadas para criação de modelos para representação de estados

mentais, crenças e objetivos. Seu objetivo é proporcionar uma linguagem de fácil entendimento para apoiar o desenvolvimento de SMA comerciais.

- *MAS-ML – Multi-Agent System Modeling Language – Linguagem de Modelagem de Sistema Multi-Agente*

Proposta por Silva (apud Guedes, 2012) a MAS-ML foi desenvolvida a partir da UML 2.1. Apresenta novas extensões de conceitos de modelagem, representados por novas meta-classes e estereótipos, considerados apropriados para capturar as características típicas de SMA. As extensões propostas incorporam os conceitos do modelo abstrato TAO, que define um amplo conjunto de conceitos orientados a agente (SILVA apud GUEDES, 2012). Utiliza uma extensão do Diagrama de Classes para modelar classes, agente, organização, ambientes e os relacionamentos entre essas entidades. Utiliza extensões do Diagrama de Sequência para representar interações entre as instâncias de SMA e as intra-ações de cada instância, para representar a execução de planos e ações de modelagem de intra-ação relacionadas a agentes, organizações e ambientes, e para ilustrar os protocolos descritos pelos papéis de agentes, e ainda pode ser utilizado para modelar os protocolos de interação de comunicação entre agentes e suborganizações. O Diagrama de atividades é estendido para representar planos e ações de agentes e organizações. Com base em notação própria, apresenta o Diagrama de Organização e Diagrama de Papel, que são próximos graficamente do Diagrama de Classes padrão da UML.

- **Metodologias de Engenharia de Software Orientada a Agentes**

- *Gaia*

Em Zambonelli (apud Guedes, 2012), a metodologia GAIA produz artefatos basicamente textuais. A metodologia não possui uma notação gráfica particular, embora recomende o uso da AUML para preencher essa lacuna. Esta metodologia demonstra pouca preocupação com a fase de elicitação e análise de requisitos.

- *MaSE – Multiagent Systems Engineering – Engenharia de Sistemas*

Multiagente

Segundo DeLoach (apud Guedes, 2012), MaSE emprega um certo número de modelos graficamente embasados a fim de descrever os tipos de agentes encontrados em um sistema e suas interfaces com outros agentes, do mesmo modo que uma definição independente de arquitetura sobre o desenho interno de tais agentes. Utiliza os Diagramas de Classe, Sequência, Caso de Uso, Estado, Implantação da UML como seus modelos gráficos.

- *MESSAGE/UML – Methodology for Engineering Systems of Software Agents – Metodologia para Engenharia de Sistemas de Agentes de Software*

Conforme Caire (apud Guedes, 2012), MESSAGE toma a UML como um ponto de partida e adiciona conceitos de entidade e relacionamento necessários para a modelagem orientada a agente. Utiliza o Diagrama de Classes para representar as relações entre os conceitos, e a visão de domínio do sistema. O Diagrama de Sequência é utilizado para representar as dependências temporais entre sub-tarefas. Diagramas de Atividades podem ser utilizados para exemplificar o modelo de análise Visão de Objetivo/Tarefa. Da AUML utiliza o Diagrama de Sequência para representar protocolos de interação e mensagens que são trocadas entre papéis. Os demais diagramas propostos possuem notação específica.

- *PASSI - (Process for Agent Societies Specification and Implementation – Processo para Especificação e Implementação de Sociedade de Agentes)*

Em Burrafato (apud Guedes, 2012), PASSI é uma metodologia para projetar e desenvolver sociedades multiagente integrando modelos de projeto e conceitos de Engenharia de Software e Inteligência Artificial usando a notação UML. O Diagrama de Caso de Uso é utilizado para descrever o Domínio do sistema. Os agentes são representados por pacotes dentro de um Diagrama de Caso de Uso. Os Diagramas de Sequência são utilizados para apresentar os detalhes dos Casos de Uso, e para representar as formas de comunicação entre agentes. O Diagrama

de Atividades apresenta o comportamento de cada agente, e também pode representar o fluxo de evento entre e dentro tanto das classes de agentes principais como de suas classes internas. O Diagrama de Classe pode representar a descrição de ontologia de comunicação do sistema, e a descrição de papéis, onde cada classe é um papel e um agente é representado por um pacote. E ainda, podem representar as visões dos agentes dentro do sistema e o sistema como um todo. Os protocolos de comunicação são descritos seguindo o padrão FIPA e, geralmente, representado por diagramas de sequência AUML. E, por fim, os diagramas de implantação são criados ao final da metodologia e ilustram a locação dos agentes, seu movimento e seu apoio de comunicação.

- *Prometheus*

De acordo com Padgham (apud Guedes, 2012) essa metodologia suporta especificamente o desenvolvimento de agentes do tipo BDI² que utilizam objetivos, crenças e planos e eventos. Os trabalhos Khallouf, Padgham, Padgham, Winikoff e Cheong (apud Guedes, 2012) apresentam modelos de aplicação e melhorias sobre a metodologia. A mais significativa está em Winikoff (apud Guedes, 2012), onde os Diagramas de Casos de Uso são substituídos por Diagramas de Cenários. A metodologia possui notação própria, e é dividida em três fases: A fase de especificação, que produz como artefatos os diagramas de Cenários, de Visão Geral de Objetivos, de Papel e de Visão Geral de Análise. A fase de projeto arquitetural produz como artefatos os diagramas de Familiaridade de Agente, de Acoplamento de Dados, de Ligação de Papel de Agente, de Visão Geral do Sistema, de Interação e de Protocolos. O Diagrama de Protocolos, utiliza a notação da AUML para especificar os protocolos de comunicação. A última fase, fase de projeto detalhado, refina os artefatos produzidos nas etapas anteriores, e tem como artefatos os Diagramas de Visão Geral de Agente, de Capacidade, de Visão Geral de Capacidade. Um exemplo de cada diagrama citado por ser visto em RMIT (apud Guedes, 2012).

2 BDI é a sigla de *belief, desire e intention*. Agentes BDI, são agentes deliberativos baseados em estados mentais. Entre os estados mentais estão crenças, desejos e intenções (Carvalho, 2004).

- *Tropos*

Em Castro e outros (apud Guedes, 2012) e com colaborações em Bresciani e outros (apud Guedes, 2012) e Giorgini e outros (apud Guedes, 2012), essa metodologia foi construída para suportar todas as fases de análise e projeto no processo de desenvolvimento de software, desde a análise de domínio da aplicação até a implementação do sistema. A metodologia possui uma notação própria e é dividida em cinco fases: Requisitos Iniciais, Requisitos Finais, Projeto Arquitetural, Projeto Detalhado e Implementação. Nas fases de requisitos iniciais e finais são produzidos modelos que representam os requisitos funcionais, requisitos não-funcionais e a relação entre os atores do sistema.

Segundo Castro, Alencar e Silva (apud Guedes, 2012), Tropos é fundamentado nos conceitos oferecidos pelo *framework* de modelagem *i** que inclui os conceitos de ator (agentes, posições ou papéis) e suas interdependências intencionais, incluindo dependências de objetivo, objetivo-*soft*, tarefa e recurso (YU apud GUEDES, 2012). O *i** possui os modelos de dependência estratégica e de razão estratégica, artefatos produzidos nas fases Requisitos Iniciais, e refinados na fase de Requisitos Finais. A fase de projeto arquitetural produz como artefato o Modelo Arquitetural do sistema. Na fase de projeto detalhado, são incorporados detalhes adicionais a cada componente do modelo arquitetural do sistema. Nesta fase, a especificação da estrutura, comunicação e comportamento são modeladas a partir das notações disponibilizadas pela AUML. O diagrama de classes da UML pode ser estendido para representar agentes do tipo BDI, em casos particulares.

Destacamos que tanto as linguagens, quanto as metodologias não são completas e não atendem, sozinhas, todos os casos para o desenvolvimento de sistemas multiagente. Dessa forma, para que as várias características e necessidades deste tipo de sistema sejam satisfeitas é aceitável que as linguagens ou metodologias sejam mescladas.

Enfim, um SMA deve ser composto por técnicas e algoritmos que serão utilizados

por seus agentes para atingir os objetivos para os quais foram projetados. A seguir apresentamos a fundamentação teórica estudada sobre Recuperação de Informações em Textos, foco computacional principal de nossa pesquisa.

3.2 RECUPERAÇÃO DE INFORMAÇÕES EM TEXTOS

A Recuperação de Informações (RI) tem como foco a busca de informações em documentos ou textos, e a classificação destas buscas a fim de retornar os documentos mais importantes em uma coleção, dada uma quantidade limitada de informações a serem resgatadas. De acordo com Manning, Raghaven e Schütze (2009), “RI busca encontrar material de natureza não estruturada que satisfaça uma necessidade de informação, a partir de grandes coleções”.

Outro importante aspecto das pesquisas em RI é a utilização do **Modelo Espaço Vetorial (SVM, do inglês *Vector Space Model*)** para representação computacional das coleções que serão analisadas (SALTON, WONG e YANG, 1975). Este modelo representa os documentos de uma coleção como vetores em um espaço multidimensional. Cada uma das dimensões desses vetores representa a frequência de uma palavra, elemento ou termo presente nos documentos correspondentes. Para uma coleção de N documentos $d=(d_1, d_2, \dots, d_N)$ que contém M elementos distintos $t=(t_1, t_2, \dots, t_M)$. Cada documento d_i é um vetor $d_i=(a_{i1}, a_{i2}, \dots, a_{iM})$, no qual o valor a_{ij} é uma medida de ocorrência, ou frequência do termo t_j no documento d_i . Para calcular a frequência ou ocorrência dos termos (a_{ij}) , aplica-se a medida TF-IDF, do inglês *Term Frequency Inverse Document Frequency* (SALTON, WONG e YANG, 1975). Esse cálculo é feito conforme a equação:

$$TfIdf(w, d) = TF(w, d) \times \log\left(\frac{N}{DF(w)}\right) \quad (3.1)$$

onde $TF(w, d)$ é a frequência de cada palavra no documento, N é o número de todos os documentos da coleção, e $DF(w)$ é o número de documentos que contém o termo w .

Nesta medida, a importância de um termo é proporcional à sua frequência de ocorrência em cada documento da coleção, e inversamente proporcional ao número de documentos em que o termo aparece. Dessarte, o vetor que representa cada documento, contém todos os termos utilizados em todos os documentos da coleção. Sendo assim, quanto maior a coleção, maior a dimensionalidade do espaço vetorial que a representa. Para isso, a aplicação de técnicas de pré-processamento na coleção que será analisada pode melhorar a qualidade de representação vetorial da mesma.

Técnicas de pré-processamento dos textos são utilizadas em conjunto com técnicas de RI, PLN ou TM, e servem para preparar os *corpus* para posterior análise. Dentre elas, a **Remoção de Termos Frequentes ou Stopwords** que tem por objetivo excluir termos que não influenciam na análise e identificação de informações importantes. Os termos frequentes, geralmente fazem parte de classes gramaticais como artigos, preposições, advérbios, pronomes e demais classes de palavras auxiliares, utilizadas para melhorar a compreensão e leitura do leitor humano, mas, que não influenciam diretamente na análise computacional do texto. Essa técnica pode ser desenvolvida com base numa lista que contenha as palavras mais frequentes de um idioma ou com base na classificação gramatical atribuída aos termos por um etiquetador morfossintático automático.

A **Etiquetagem Morfossintática** (do inglês *Part-of-speech Tagging*) é uma técnica de pré-processamento de textos importante em aplicações de extração e recuperação de informações. Essa técnica, ligada à PLN, visa atribuir aos termos analisados, etiquetas relativas às suas categorias gramaticais, tais como: substantivos, verbos, adjetivos, artigos, etc. O etiquetador automático é a ferramenta computacional desenvolvida para auxiliar na realização desta tarefa.

O etiquetador automático explorado nessa pesquisa, faz parte do projeto NLPNet³ (FONSECA; ROSA, 2013), construído com base no conjunto de etiquetas do *corpus* Mac-Morpho⁴ e é capaz de atribuir ao termo analisado uma das seguintes etiquetas: Artigo – ART; Adjetivo – ADJ; Nome – N; Nome Próprio – NPROP; Numeral – NUM; Pronome Adjetivo – PROADJ; Pronome Substantivo – PROSUB; Pronome Pessoal –

3 Disponível em <http://nilc.icmc.usp.br/nlpnet/> - acesso em 11.12.2015

4 Disponível em <http://nilc.icmc.usp.br/macmorpho/> - acesso em 11.12.2015

PROPESS; Pronome conectivo subordinativo – PRO-KS; Pronome conectivo subordinativo relativo – PRO-KS-REL; Advérbio – ADV; Advérbio conectivo subordinativo – ADV-KS; Advérbio relativo subordinativo – ADV-KS-REL; Conjunção coordenativa – KC; Conjunção subordinativa – KS; Preposição – PREP; Interjeição – IN; Verbo – V; Verbo auxiliar – VAUX; Particípio – PCP; Palavra denotativa – PDEN; Pontuação – PU, etc (ALUÍSIO et al., 2003).

Abaixo apresenta-se um exemplo de aplicação do etiquetador automático, na qual o segundo valor da tupla é referente a categoria gramatical de cada palavra da frase.

Exemplo:

Frase: "um outro exemplo disso é obra quem mexeu no meu queijo"

Frase após a aplicação do etiquetador: (u'um', u'ART'), (u'outro', u'PROADJ'), (u'exemplo', u'N'), (u'disso', u'PREP+PROSUB'), (u'\xe9', u'V'), (u'obra', u'N'), (u'quem', u'PRO-KS'), (u'mexeu', u'V'), (u'no', u'PREP+ART'), (u'meu', u'PROADJ'), (u'queijo', u'N')

A partir da aplicação do etiquetador automático, todos os termos analisados passam a ser relacionados junto à sua classe gramatical, o que auxilia, por exemplo, no processo de identificação de termos sinônimos ou na identificação de ambiguidades. Além disso, a partir da análise das etiquetas associadas aos termos é possível reduzir o *corpus* de análise ao ignorar ou mesmo excluir termos classificados como artigos e/ou preposições, uma vez que estes termos estão presentes nos textos para facilitar o entendimento e leitura humana, e não são relevantes para a definição do contexto e/ou informações relevantes do texto.

Segundo Manning, Raghavan e Schütze (2009), a sinonímia, ou seja, a representação de um mesmo conceito por termos diferentes causa impacto na revocação⁵ (*recall*) em sistemas de RI. Duas formas são utilizadas para resolução deste problema: Métodos Globais ou Métodos Locais. Dentre os métodos globais, está a Consulta de Expansão através de um dicionário de sinônimos ou *Wordnet*.

O idioma Português, idioma principal dos textos analisados pela solução proposta por essa pesquisa, possui uma grande quantidade de palavras sinônimas, ou seja, palavras diferentes que em determinados contextos possuem o mesmo significado. Para auxiliar a análise dos termos contidos no *corpus*, o método de **Normalização**

5 Métrica utilizada para medir o número de documentos relevantes recuperados em relação do número total de documentos de uma coleção.

foi desenvolvido com base no dicionário de sinônimos disponibilizado pelo Projeto TeP2.0 (DIAS-DA-SILVA et al., 2000).

O TeP2.0 é uma coleção de palavras do português agrupadas em conjuntos que mantêm uma relação de sinonímia (sinônimos) entre si. Portanto, em cada conjunto, as palavras têm um mesmo significado. (TeP2.0 beta, acesso em 12 de dezembro de 2015)

Dessa forma, termos que constam num mesmo grupo de relação sinonímica são representadas por um único termo. Este termo, para fins desta pesquisa, é considerado termo canônico e representa o conjunto de termos sinônimos.

Outra técnica de pré-processamento do texto é a **Redução ao Radical** ou, em inglês, **Stemming**. Segundo Orenge e Huyck (2001), a Redução ao Radical ou *Stemming*, visa a decomposição de formas variantes de uma palavra em uma representação comum, o radical (*stem*). Neste sentido, baseado nos trabalhos de Porter (apud ORENCO; HUYCK, 2001), Orenge e Huyck (2001) apresentam a construção de um *stemmer* baseado em regras, desenvolvido exclusivamente para tratamento das características intrínsecas ao idioma Português, o RSLP – Removedor de Sufixos para Língua Portuguesa. O RSLP remove os sufixos dos termos e retorna apenas o radical de formação de cada termo. Dessa forma, os termos constantes no *corpus* serão reduzidos ao seu radical de formação auxiliando assim a procura por termos similares e/ou semelhantes.

De acordo com Dumais (2004), **Análise Semântica Latente** (LSA, do inglês Latent Semantic Analysis):

[...] é uma abordagem estatística totalmente automática utilizada para extrair relações entre palavras por meio de seus contextos de uso em documentos, passagens ou frases. Não faz uso de técnicas de processamento de linguagem natural para a análise morfológica, sintática, ou relações semânticas. (DUMAIS, 2004, p. 191, tradução nossa)

Além disso, a LSA é considerada uma técnica de aprendizado de máquina não supervisionado. Através de sua análise, puramente estatística, são criadas matrizes de relações entre os termos e os documentos. Na matriz que representa o *corpus* as palavras são representadas pelas linhas e as colunas correspondem aos textos que compõem o *corpus*. A matriz é preenchida com a frequência absoluta de cada palavra, em cada uma de suas entradas na matriz, em seguida o valor é convertido para o seu correspondente logarítmico. Isto é feito baseando-se no fato de que um

documento com, por exemplo, três ocorrências de uma mesma palavra, tende a ser mais importante do que um documento com apenas uma ocorrência, porém não três vezes mais importante. Em seguida, cada um dos novos valores de entrada é dividido pelo somatório do produto destes valores pelo logaritmo dos mesmos, para salientar a sua importância.

Após o cálculo da frequência, é aplicada a técnica SVD – Decomposição de Valor Singular, sobre as matrizes. Essa técnica é utilizada para localizar a informação semântica essencial em uma matriz de coocorrência de palavras.

Com a aplicação do SVD, são criadas três novas matrizes (A, B, C) a partir da matriz original, e a matriz resultante (M) é composta pelo produto entre as 3 matrizes novas $(M = A \times B \times C)$. A matriz A é composta pelos vetores singulares à esquerda, a matriz B é composta pela diagonal de valores singulares em ordem decrescente, e a matriz C é composta pelos vetores singulares à direita. Com a aplicação do SVD as dimensões da matriz são reduzidas, mantendo apenas os maiores valores singulares.

O objetivo final da decomposição é que a matriz M contenha um espaço semântico condensado que representa as melhores relações entre as palavras e os documentos. A proximidade entre as palavras é calculada através do cosseno do ângulo entre os seus vetores. Quanto maior o cosseno do ângulo, mais próximas as palavras são.

A LSA também pode ser considerada como um modelo de representação de dados ou textos, uma vez que o resultado final de aplicação da LSA, é uma matriz de representação da frequência e importância das palavras dentro de um *corpus*, que considera a proximidade entre duas palavras e sua possível relação semântica.

A partir destas análises busca-se a descoberta de estruturas semelhantes que podem auxiliar a recuperação de informações relevantes. Outra característica da LSA é não considerar a ordem dos termos nos documentos, tipificando uma abordagem com o modelo *bag-of-words*⁶.

6 O modelo *bag-of-words* descarta todas as informações que lhe são comunicadas pela ordem das palavras em frases escritas em linguagem natural (MANNING, RAGHAVAN e SCHÜTZE, 2009, p. 269, tradução nossa).

3.2.1 Clusterização

Em outro viés das pesquisas em RI, MT e PLN, encontram-se os Algoritmos de Agrupamento ou Clusterização, que são utilizados para, a partir da análise de características semelhantes entre os textos que compõem o *corpus*, criar grupos de semelhança, onde pertencentes a um mesmo grupo possuem um maior nível de semelhança entre eles do que com os membros de outros grupos. Em RI, algoritmos de clusterização são utilizados para agrupar automaticamente os resultados recuperados em uma busca, facilitando assim a identificação de diversos documentos que atendem aos termos da busca.

Segundo Manning, Raghaven e Schütze (2009) a **Clusterização** é a forma mais comum de aprendizado não supervisionado, ou seja, não há interferência humana no processo. Além disso, os algoritmos de clusterização podem ser classificados como Particional ou Hierárquicos. Os algoritmos de clusterização particional criam uma quantidade pré-determinada de grupos com características semelhantes, mas não seguem uma estrutura definida. Os algoritmos hierárquicos são mais informativos e organizam os grupos de forma hierárquica, usualmente na estrutura de árvore. Além disso não é necessário a definição da quantidade de grupos que devem ser criados com sua aplicação. Contudo, os algoritmos de clusterização hierárquicos são custosos computacionalmente, possuem baixa eficiência e geralmente possuem complexidade quadrática ao número de documentos que serão processados. Enquanto os algoritmos de clusterização particional possuem complexidade linear (MANNING, RAGHAVAN e SCHÜTZE, 2009).

O Algoritmo **K-Means (KM)** é um dos mais importantes algoritmos de clusterização particional. Para utilização do KM é necessário que o *corpus* seja previamente tratado e esteja representado por um modelo. Dessa forma, o KM particiona o *corpus* em vetores de documentos, e com o uso da Distância Euclidiana, como métrica para análise da semelhança entre os vetores que representam os termos dos documentos e o centróide do grupo, cria os agrupamentos. Por ser um algoritmo de clusterização particional, tem como característica a predefinição da quantidade de grupos que devem ser criados. Outra característica importante do KM é a realização de n iterações até a estabilização dos membros de um grupo, por

consequência cada iteração busca maximizar as semelhanças dos membros de um mesmo grupo e distanciá-los de outros grupos (MANNING, RAGHAVAN e SCHÜTZE, 2009).

O pseudoalgoritmo do KM é apresentado abaixo (adaptado de MANNING, RAGHAVAN e SCHÜTZE, 2009, p. 361):

```
KMeans({x1, x2, ..., xn}, K, A) #entrada
(c1, c2, ..., cK) ← cria_centroides({x1, x2, ..., xn}, K) #criando K
centróides
  Para cada grupo K, faça:
  uk ← ck #atribuindo cada centróide a um grupo
  enquanto o critério de parada não for atingido, faça: #enquanto
  houver modificações entre os membros dos grupos
    para cada grupo K, faça:
      grupo[k] = {}
    para cada ponto P, faça: #o total de pontos é n
      atribui_ponto_ao_grupo(P, A, grupo) #calcula-se a
  distância do ponto ao centróide de cada grupo e
  este ponto permanecerá no grupo que tiver a menor
  distância
  para cada grupo K, faça:
    uk ← novo_centroide(K) #recalcula os centróides dos grupos
  retorna {u1, u2, ..., uk}
```

Evidencia-se que neste caso, a escolha inicial dos centróides é aleatória e que o número de centróides é igual ao número de grupos que serão formados. Além disso, a complexidade do KM é linear e pode ser expressa pela equação $O(n \times K \times I \times d)$, onde n é o total de vetores, K é o número predefinido de grupos, I equivale ao número de iterações e d o número de atributos e/ou características.

Ademais, como dito anteriormente, o KM utiliza como métrica a **Distância Euclidiana** (DE), que segundo Feldman e Sanger (2007) é a métrica mais popular e explorada em pesquisas de RI. A DE é representada pela equação:

$$DE(x_1, x_2) = \sqrt{\sum_k (x_{1k} - x_{2k})^2} \quad (3.2)$$

A DE resulta na diferença entre as distâncias de dois vetores, desta forma, quanto mais próximo de zero for o valor da distância, mais similares são os documentos.

Contudo, ainda segundo Feldman e Sanger (2007), a **Similaridade de Cossenos** é a medida mais comumente utilizada para a clusterização de textos. A Similaridade de Cossenos é representada pela equação:

$$Scos(x_i, x_j) = (x'_i \cdot x'_j) = \sum_k x'_{ik} \cdot x'_{jk} \quad (3.3)$$

onde x' é o vetor de documentos normalizado, $x = x/|x|$. O resultado apresentado pela similaridade de cossenos é a medida do ângulo entre os vetores analisados, entre $[0,1]$. Quanto mais próximo de 1, mais similares são os termos analisados.

3.3 CONSIDERAÇÕES FINAIS DO CAPÍTULO

Neste capítulo apresentamos os estudos realizados na literatura científica e acadêmica a fim de adquirirmos conhecimento teórico suficiente para embasar o desenvolvimento da proposta de solução que será apresentada no Capítulo 5.

Apresentamos nossas considerações sobre as principais linguagens e metodologias para o desenvolvimento de Sistemas Multiagente. Trazendo os conceitos, modelos e características principais das linguagens derivadas da tradicional UML. Além de apresentarmos as metodologias de engenharia de software orientada a agentes que consideramos de maior importância.

Ainda expomos as principais técnicas utilizadas para realizar a Recuperação de Informações em Textos, tendo como ponto central destes estudos foi a busca por métodos, técnicas e algoritmos que pudessem auxiliar na análise de textos curtos sobre qualquer área do conhecimento produzidos em Ambientes Virtuais de Aprendizagem.

4 TRABALHOS CORRELATOS

Neste capítulo apresentamos os trabalhos encontrados na literatura acadêmica que identificamos ter como objetivo realizar a Recuperação de Informação em Textos Curtos para auxiliar o professor em uma de suas atividades cotidianas.

Na Seção 4.1 fazemos um breve relato de como os trabalhos foram selecionados e detalhamos os oito que julgamos conter maior proximidade aos objetivos destacados por esta pesquisa.

Na Seção 4.2 destacamos as semelhanças e diferenças entre nosso trabalho e os trabalhos selecionados e descritos.

Por fim, na Seção 4.3 trazemos as considerações finais do capítulo.

4.1 LEVANTAMENTO DO ESTADO DA ARTE

Durante a elaboração desta pesquisa realizamos uma investigação na literatura científica, mais especificamente em artigos publicados em revistas e anais de eventos, entre os anos de 1992 e 2015. As buscas basearam-se, primeiramente na seleção de palavras chaves relacionadas à Recuperação de Informações em Textos, que depois foi refinada a fim de encontrarmos trabalhos que tratassem especificamente da recuperação de informações em textos curtos e da recuperação de Informações em ambientes virtuais de aprendizagem.

Realizadas as buscas, selecionamos, através da leitura dos resumos dos artigos, aqueles que continham técnicas e/ou conceitos que pudessem nos nortear. Essa busca resultou na catalogação de 119 artigos. Destes 119, selecionamos 40 artigos para uma análise mais detalhada. O critério de escolha foi a relevância da temática de cada artigo em relação à temática central dessa pesquisa, ou seja, se os artigos eram relacionados ao contexto de aplicação de informática para apoiar a educação, se os textos analisados eram produzidos em ambientes virtuais de aprendizagem, e se o objetivo dos trabalhos era apoiar o professor em suas atividades.

O estudo detalhado dos 40 artigos foi direcionado para obtermos a resposta de cinco

questões principais:

- Qual a ideia principal do artigo?
- Quais os aspectos positivos?
- Quais os aspectos negativos?
- Qual a semelhança entre o artigo e sua proposta?
- Quais questões ou ideias foram originadas com a leitura deste artigo?

De posse destas respostas, selecionamos 8 artigos que julgamos conter maior proximidade aos nossos objetivos de pesquisa. O critério de escolha foi estabelecido com base no referencial teórico descrito no Capítulo 3 e o referencial teórico descritos nos trabalhos. Além disso termos considerados como satisfatórios os resultados apresentados nos artigos em relação ao apoio oferecido ao professor.

A seguir apresentamos um breve resumo dos 8 artigos selecionados, com destaque para as principais características de cada trabalho.

1. *Automatic Assessment of the Content of Essays Based on Course Materials*

Kakkonen e Sutinen (2004) utilizam uma abordagem baseada em Análise Semântica Latente (LSA) para auxiliar no processo de correção automática de respostas discursivas num ambiente específico do tipo CAA – *Computer Assisted Assessment*. Os autores utilizam os materiais didáticos do curso para, com auxílio da LSA, criar a base de conhecimento que auxiliará no processo de correção das respostas. Eles ainda assumem que essa prática é viável, pois, basicamente, o conhecimento de um aluno é construído através da leitura do material didático indicado, e isso pode causar uma determinada similaridade semântica entre o material didático e a resposta escrita por um aluno que tenha utilizado este material para estudo (KAKKONEN e SUTINEN, 2004). Outro ponto de destaque é a utilização de respostas realizadas em semestres anteriores, que já foram analisadas e pontuadas, tanto pelo sistema quanto pelo especialista humano. As respostas com maiores notas, são utilizadas pelo módulo de LSA do sistema proposto pelos autores para criação de uma pequena base de conhecimento, que é empregada no treinamento do sistema para posterior correção das novas respostas. Outro

ponto importante é a realização das análises de similaridade entre as respostas dos alunos com trechos pré-selecionados dos livros didáticos. Por fim, Kakkonen e Sutine (2004) destacam que os resultados alcançados com seu experimentos são bons, comparados a outros sistemas CAA que utilizam a LSA. Além disso, os resultados dos experimentos comprovam que a utilização da LSA para automatizar o processo de correção das respostas discursivas é comparável com as correções realizadas por especialistas humanos.

2. Análise das Mensagens de Fóruns de Discussão através de um Software para Mineração de Textos

Azevedo, Behar e Reategui (2011) apresentam a aplicação do software MineraFórum⁷, que realiza uma análise qualitativa das mensagens enviadas por alunos em fóruns de discussão. O MineraFórum atribui um valor de relevância entre 0 e 5 para cada postagem realizada pelos alunos. Os critérios utilizados pelo software para o cálculo da relevância de uma postagem são: análise temática da mensagem, quantidade de citações da mensagem, similaridade da mensagem com outras do fórum. Um diferencial deste trabalho está na mineração de textos utilizando grafos. Dessa forma, os grafos apresentam os termos com maior ocorrência no texto, e identifica se elas estão próximas. “As associações entre os nós do grafo, palavras que mais ocorrem, indicam a proximidade entre as palavras” (AZEVEDO, BEHAR e REATEGUI, 2011, p. 21). Os experimentos foram realizados em três AVEA's diferentes e os resultados de aplicação do MineraFórum foram comparados aos resultados apresentados pelos especialistas (professores). Os autores destacam que a média das análises realizadas pelo sistema e pelos professores é semelhante, dessa forma, o MineraFórum alcançou seus objetivos. Além disso, o programa utiliza parâmetros específicos para realizar a análise, já as análises realizadas pelos professores não possuem critérios específicos (AZEVEDO, BEHAR e REATEGUI, 2011, p. 21).

7 Disponível em <http://plataforma.nie.iff.edu.br/mineraforum/> (Acesso em 20 de dezembro de 2015)

3. *Automated Assessment of Short Free-Text Responses in Computer Science using Latent Semantic Analysis*

Klein, Kyrilov e Tokman (2011) apresentam uma ferramenta para correção automática de respostas discursivas. De acordo com os autores, o foco principal é na precisão das respostas, considerando o material didático como base para a avaliação. Os algoritmos que compõem o sistema proposto são baseados em LSA e em técnicas de clusterização (KLEIN, KYRILOV e TOKMAN, 2011, p. 159). De acordo com Klein, Kyrilov e Tokman (2011), os destaques de seu trabalho estão na ausência de uma base de conhecimento pré-definida para auxiliar nas análises. Além disso, a configuração do sistema para correção automática das respostas demonstrou-se como a tarefa mais complexa do trabalho. Sendo que os resultados alcançados com a configuração correta obtiveram uma taxa de semelhança de 80% comparadas com as correções realizadas por especialistas humanos. Além disso, os autores destacam que a utilização de TF-IDF superou suas expectativas, e que a utilização do algoritmo de clusterização K-Means alcançou resultados razoáveis.

4. *Automatização do Processo de Identificação de Presença Social em Fóruns e Chats*

Silva e outros (2012) apresentam uma proposta de ferramenta para auxiliar no processo de análise de Presença Social (PS) do aluno dentro do AVEA. A PS é “o grau de sentimento, da percepção de pertencimento no grupo, na comunidade em interação, e reação ao conectar-se com outros indivíduos por recursos de comunicação mediada por computador” (SILVA et al. apud TU, 2012). Os autores destacam que a PS é composta por elementos que caracterizam de fato as interações do aluno o que comprova que ele “esteve presente” no AVEA e por isso é distinta das informações sobre o registro de login e o histórico de acesso às ferramentas do ambiente. Um destaque do trabalho é o Construtor de Categorias, módulo que é configurado pelo professor ou tutor com as categorias e subcategorias de presença social, e também pistas textuais. A PS é identificada a partir da análise das interações realizadas pelos alunos através da produção de textos em fóruns ou chats.

Estes textos são recuperados e transformados para um formato específico que facilitará sua análise pelo Analisador. O Analisador é responsável por processar o texto, buscando identificar características pertinentes às categorias pré-definidas. Silva e outros apresentam os resultados de seus experimentos realizados, que foram verificados e validados por um especialista. Em um dos experimentos os resultados automáticos chegaram em 94% de acerto, comparadas à análise realizada pelo especialista.

5. *Openanswer: A Framework to Support Teacher's Management of Open Answers Through Peer Assessment*

Sterbini e Temperini (2013) apresentam um ambiente *web* para análise automatizada de respostas discursivas, denominado OpenAnswer. Para realizar a análise, é necessário que os professores cadastrem possíveis respostas no sistema. Essas respostas serão utilizadas como parâmetros para análise de similaridade das respostas inseridas pelos alunos. Para calcular a similaridade, o sistema faz uso de um módulo desenvolvido com base nas Redes Bayesianas. São feitas classificações em todas as respostas analisadas, e através dessas classificações (boa, razoável, ruim) o aluno sabe se sua resposta está dentro do tema da pergunta. A análise das respostas leva em consideração a similaridade entre o texto inserido pelo aluno, o texto de resposta padrão inserido pelo professor, e a similaridade com as respostas dos outros alunos. Os resultados apresentados são considerados pelos autores como razoáveis, uma vez que o sistema consegue corrigir cerca de 30% das respostas a ponto de não ser necessário que o professor as confira. Sendo assim, a utilização do sistema reduz em 30% o trabalho do professor (STERBINI; TEMPERINI, 2013). Da mesma maneira, o sistema pode ser considerado como genérico, pois ao inserir as perguntas o professor deve inserir também padrões de respostas que serão utilizados como base para realizar as correções das respostas enviadas pelos alunos.

6. *Uso de Técnicas de Pré-Processamento Textual e Algoritmos de Comparação como Suporte à Correção de Questões Dissertativas: Experimentos, Análises e Contribuições*

Ávila e Soares (2013) apresentam um sistema para análise de respostas discursivas elaboradas no idioma Português. Para tratar as peculiaridades do idioma, os autores realizaram adaptações nos algoritmos Força-Bruta (ÁVILA; SOARES apud ZIVIANI, 2010), Boyer-Moore (ÁVILA; SOARES apud BOYER; MOORE, 1977), KMP (ÁVILA; SOARES apud KNUTH; MORRIS; PRATT, 1977), Levenshtein (*Edit Distance*) (ÁVILA; SOARES apud LEVENSHTAIN, 1966) e Rabin-Karp (ÁVILA; SOARES apud KARP; RABIN, 1987) que, por padrão analisam apenas palavras (*strings*), para que a análise realizada considera pequenas frases (ÁVILA; SOARES, 2013). Foram implementadas as seguintes técnicas de pré-processamento do texto: substituição de caracteres acentuados pelo correspondente sem acento; exclusão de palavras repetidas; alteração de todos os caracteres para maiúsculo; remoção das palavras frequentes; redução do radical (*stemming*); e normalização de termos. Como destaque, os autores apresentam uma nova técnica denominada SSD – Substituir Sequência Pré-definida. Essa técnica foi desenvolvida para verificar se a inversão de palavras dentro de uma sentença, como por exemplo, “emissor para o receptor” e “receptor para o emissor” poderia influenciar no índice da taxa de similaridade (ÁVILA e SOARES, 2013). A base utilizada para o cálculo de similaridade das respostas é alimentada com mais de um padrão de resposta para cada pergunta, e pode ser realimentada com respostas de outros alunos a fim de criar uma aproximação entre os vocabulários utilizados pelos alunos e assim aumentar o grau de similaridade nas correções. Os autores destacam como principal contribuição do trabalho a validação e avaliação dos algoritmos de comparação de textos combinados com as técnicas de pré-processamento textual. E destacam que os resultados obtidos validam a proposta e uso desta em AVEA para auxiliar no processo de correção de respostas discursivas.

7. *Scaffolding Student Online Discussions Using Past Discussions: Pedabot Studies*

Kim e Shaw (2014) apresentam um estudo de aplicação do PedaBot, que é uma ferramenta para mineração dos textos escritos em fóruns do estilo pergunta-resposta e tem por objetivo fazer sugestões de textos e/ou outros

tópicos de discussão sobre o assunto pesquisado pelo aluno. O PedaBot é composto por módulos que são responsáveis pelo pré-processamento dos textos, extração dos termos relevantes e criação do glossário, identificação de tópicos relacionados e postagens relevantes de cada tópico. Para criação do glossário e extração dos termos relevantes são utilizados livros sobre o tópico em discussão. O sistema foi desenvolvido com a combinação de técnicas de LSA, TF-IDF, técnicas de processamento de linguagem natural e recuperação de informações. A métrica utilizada para calcular a similaridade entre os textos analisados é a Similaridade de Cossenos (SC), sendo que a SC é aplicada tanto sobre o espaço vetorial criado com a aplicação de TF-IDF, quanto sobre o espaço multidimensional criado com a aplicação de LSA. De posse desses dois resultados, calcula-se a média entre eles, e a partir da média é feita uma classificação decrescente dos valores de similaridade e os três mais similares são apresentados como sugestões para o aluno (KIM e SHAW, 2014). Os resultados apresentados por Kim e Shaw (2014) mostram que a utilização de LSA e TF-IDF é proveitosa, uma vez que os resultados medidos em relação à coerência das sugestões realizadas pelo sistema chegam a 76,7% de acerto. Além disso, os autores consideram que as indicações realizadas facilitam a construção do conhecimento dos alunos.

8. *Socially Augmented Argumentation Tools: Rationale, Design and Evaluation of a Debate Dashboard*

landoli e outros (2014) apresentam o Sistema *Debate Dashboard*, concebido através da integração com uma ferramenta já existente para mapeamento de argumentos (Cohere). O objetivo do trabalho é contribuir para o debate em torno do uso e design de tecnologias *web* de argumentação para apoiar tarefas de conhecimento distribuídas tais como os grupos de deliberação. No artigo são apresentados argumentos e evidências empíricas que mostram que as ferramentas para ambientes colaborativos com suporte computacional para visualização de argumentações podem ser desenvolvidas para apresentar mapas conceituais sobre as interações realizadas pelo participante com o objetivo de fornecer a cada um a possibilidade de acompanhar o seu desenvolvimento dentro do debate, quais foram suas

contribuições, quais as relações foram realizadas com os argumentos feitos por ele. Importante ressaltar, que o *Debate Dashboard* visa auxiliar na construção de deliberações coletivas, ou seja, dar suporte para retirada de afirmações relevantes para o tema em debate que sejam suficientes para criar um entendimento coletivo do que foi discutido. Além disso, apresenta para o usuário suas informações de interação com outros usuários e possibilitar a comunicação entre eles, podem gerar melhores oportunidades para a construção do conhecimento de um indivíduo. Além disso, ferramentas visuais, como mapas, são importantes para demonstrar de forma mais clara e objetiva as conexões de um usuário com os demais.

4.2 AS RELAÇÕES ENTRE OS TRABALHOS CORRELATOS E A PROPOSTA DESTA DISSERTAÇÃO

Como pode ser observado, os trabalhos descritos possuem pelo menos uma das técnicas, métodos ou algoritmos apresentados no Capítulo 3. Igualmente, todos os trabalhos foram desenvolvidos para auxiliar o professor de alguma maneira, seja para identificar se um aluno está de fato participando das atividades propostas através de sua Presença Social num AVEA, ou para verificar se um aluno está colaborando com a discussão proposta num fórum, ou ainda ajudar na correção de respostas discursivas.

Diante disso, no Quadro 1 apresentamos nossas considerações sobre as principais semelhanças e diferenças entre os 8 trabalhos descritos e a proposta de solução que será apresentada no Capítulo 5.

Quadro 1 - Semelhanças e Diferenças – Trabalhos Correlatos x Proposta

Trabalho correlato	Semelhança	Diferenças
<i>Automatic Assessment of the Content of Essays Based on Course Materials</i> (KAKKONEN; SUTINEN, 2004)	<ul style="list-style-type: none"> - Uso de LSA - Uso do Material Didático para apoiar a análise de similaridade 	<ul style="list-style-type: none"> - Sistema desenvolvido para correção de respostas discursivas - Utiliza uma base de conhecimento para auxiliar nas correções
Análise das Mensagens de Fóruns de Discussão através de um Software para Mineração de Textos (AZEVEDO; BEHAR; REATEGUI, 2011)	<ul style="list-style-type: none"> - Tratamento de textos produzidos em Português - Utilização de técnicas de pré-processamento 	<ul style="list-style-type: none"> - Usa Técnicas de Mineração de textos com apoio de grafos
<i>Automated Assessment of Short Free-Text Responses in Computer Science using Latent Semantic Analysis</i> (KLEIN; KYRILOV; TOKMAN, 2011)	<ul style="list-style-type: none"> - Uso do Material Didático para apoiar a análise de similaridade - Ausência de uma base conhecimentos pré-definida - Uso de TF-IDF - Uso do Algoritmo de K-Means 	<ul style="list-style-type: none"> - Sistema desenvolvido para correção de respostas discursivas
Automatização do Processo de Identificação de Presença Social em Fóruns e Chats (SILVA et al., 2012)	<ul style="list-style-type: none"> - Tratamento de textos produzidos em Português - Utilização de técnicas de recuperação de informações 	<ul style="list-style-type: none"> - Foco na identificação da Presença Social - Utiliza “pistas textuais” informadas pelos usuários para auxiliar na análise
<i>Openanswer: A Framework to Support Teacher's Management of Open Answers Through Peer Assessment</i> (STERBINI; TEMPERINI, 2013)	<ul style="list-style-type: none"> - Sistema capaz de analisar textos inseridos sobre qualquer área de conhecimento 	<ul style="list-style-type: none"> - Desenvolvido para auxiliar na correção respostas discursivas - Utiliza Redes Bayesianas para auxiliar no cálculo de similaridade
Uso de Técnicas de Pré-Processamento Textual e Algoritmos de Comparação como Suporte à Correção de Questões Dissertativas: Experimentos, Análises e Contribuições (ÁVILA; SOARES, 2013).	<ul style="list-style-type: none"> - Tratamento de textos produzidos em Português - Uso de técnicas de pré-processamento textual - Uso de Redução ao Radical - Uso de Normalização de Termos 	<ul style="list-style-type: none"> - Sistema de análise de respostas discursivas - Foco na comparação dos resultados obtidos com a aplicação de vários algoritmos para análise de similaridade entre palavras
<i>Scaffolding Student Online Discussions Using Past Discussions: PedaBot Studies</i> (KIM; SHAW, 2014)	<ul style="list-style-type: none"> - Uso de LSA - Uso de TF-IDF - Uso de Similaridade de Cossenos - Uso de material sobre o assunto para auxiliar na análise de similaridade 	<ul style="list-style-type: none"> - Utiliza um fórum do estilo pergunta-resposta como base, assim, comporta-se basicamente como um sistema para correção de respostas discursivas
<i>Socially Augmented Argumentation Tools: Rationale, Design and Evaluation of a Debate Dashboard</i> (IANDOLI et al. 2014)	<ul style="list-style-type: none"> - Sistema de suporte ao processo de debate - Utiliza técnicas de Recuperação de Informações 	<ul style="list-style-type: none"> - Tem como objetivo apresentar para o aluno como está sua participação do debate - Usa Mapas conceituais para apresentar de modo gráfico as afirmações que resgata a partir das discussões do debate - Possui uma dinâmica de debate tradicional, o que difere da dinâmica proposta pela APDT

Fonte: Autoria Própria

4.3 CONSIDERAÇÕES FINAIS DO CAPÍTULO

O trabalho mais próximo de nosso contexto, *Debate Dashboard* (IANDOLI et al. 2014) foi desenvolvido para simular a dinâmica de um debate regular, e tem por objetivo apresentar de forma gráfica para o aluno quais são suas contribuições no processo de debate e resgatar afirmações que possam ajudar no momento de fechamento do debate. O foco da solução é o aluno e não o professor/mediador.

Dentro de nossa proposta, nosso objetivo é criar suporte para auxiliar o professor enquanto mediador em um debate. E ainda, o debate por nós analisado segue uma dinâmica específica proposta pela APDT. Cabe ressaltar que os trabalhos já desenvolvidos sobre a referida AP tiveram como foco a transposição de seus conceitos para um ambiente computacional (Seção 2.2.1).

Em consequência, nosso foco de análise foram os resultados obtidos com a aplicação das técnicas, métodos e algoritmos correlacionados à fundamentação teórica estudada. Diante do exposto até aqui, elaboramos uma solução computacional que tem por objetivo fornecer suporte a análise dos textos produzidos nas interações da ADPT a fim de fornecer subsídios para a aplicação das mediações pedagógicas.

5 DESCRIÇÃO DA SOLUÇÃO

Nossa proposta de apoio Computacional à Mediação de um debate no contexto da APDT consiste na modelagem de uma arquitetura multiagente que busca oferecer apoio ao mediador em diferentes aspectos da realização de um debate, sendo alguns deles no contexto do trabalho de coordenação do debate e outros no contexto da mediação pedagógica em si.

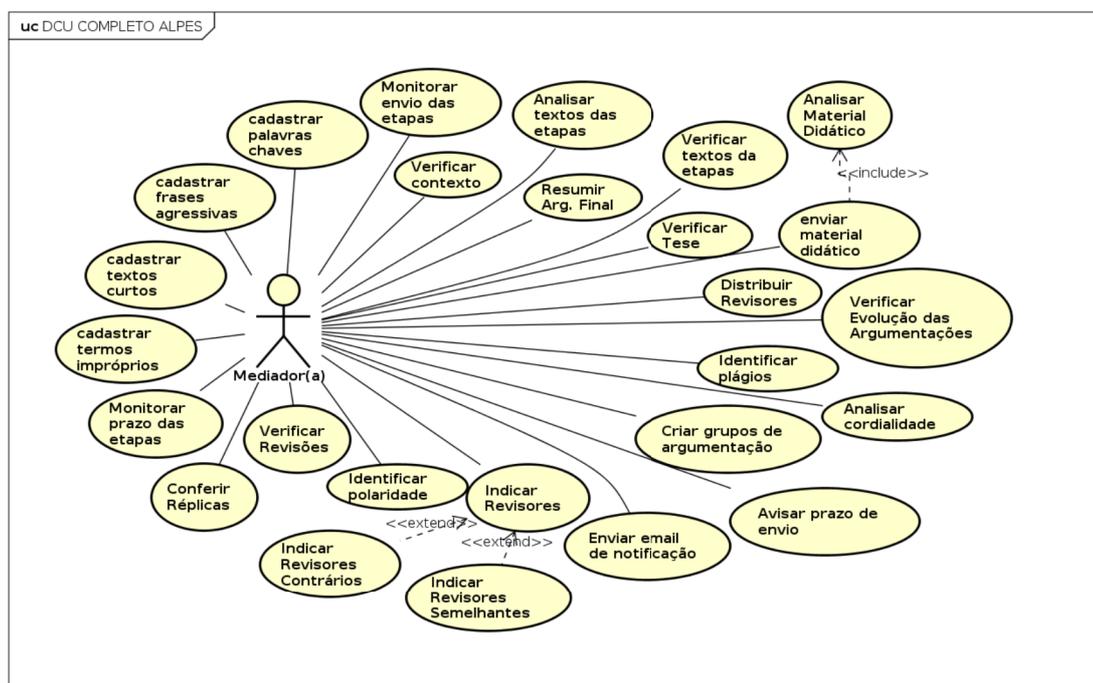
Especificamente no que diz respeito à mediação pedagógica, nosso trabalho busca oferecer recursos de processamento de texto que apoiem o trabalho no mediador na análise das produções textuais de cada participante.

Na Seção 5.1 detalhamos os elementos que compõem a Arquitetura Geral da Solução. Na Seção 5.2 mostramos a elaboração do Núcleo de Recuperação de Informações. Por fim, na Seção 5.3 apresentamos as considerações finais deste capítulo.

5.1 ARQUITETURA GERAL DA SOLUÇÃO

A solução descrita nesta Seção foi concebida a partir da análise das mediações pedagógicas descritas no Capítulo 2, com o objetivo principal de facilitar a análise das produções realizadas durante as interações da APDT. O DCU – Diagrama de Caso de Uso (Figura 5) apresenta todas as funcionalidades identificadas.

Figura 5 - Diagrama de Caso de Uso da Solução

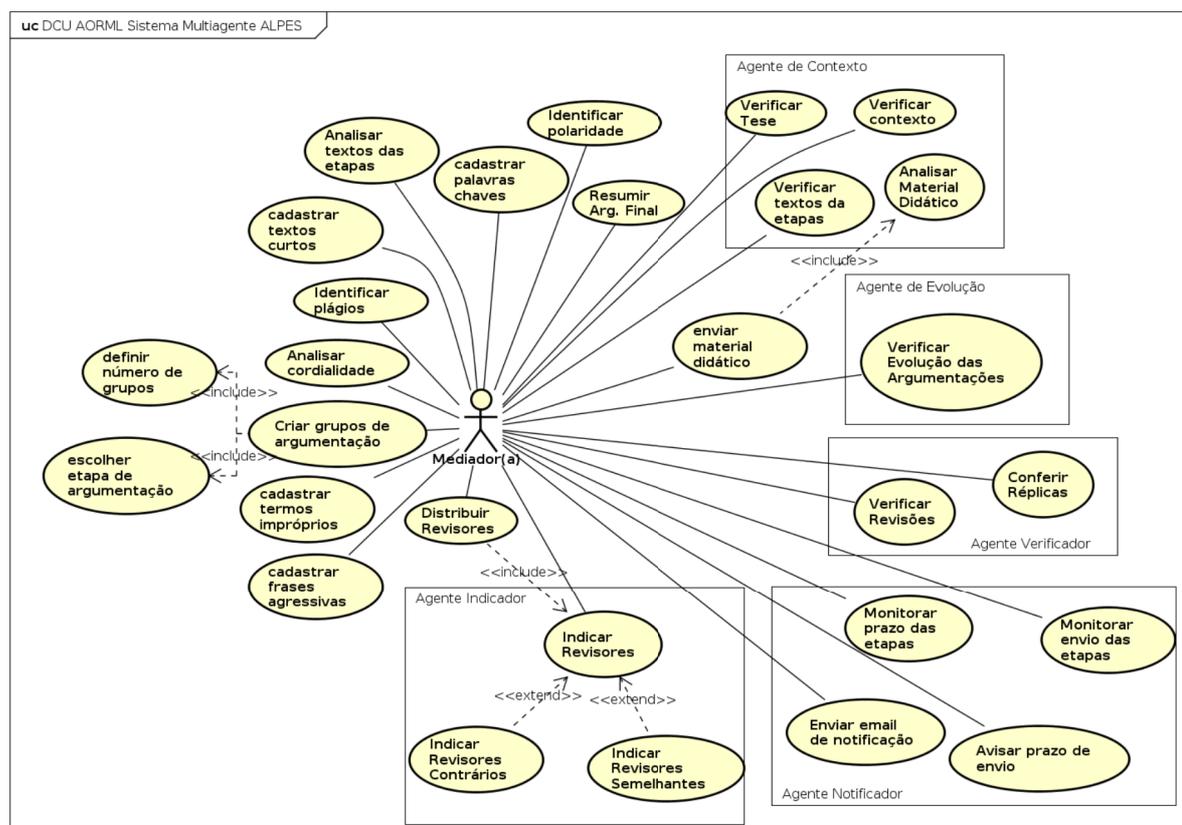


Fonte: Autoria Própria

A partir da análise deste diagrama e das funcionalidades requeridas, a solução foi elaborada como um Sistema Multiagente, denominado ALPES, no qual os agentes que a compõem são reativos e tem como domínio de atuação o Sistema “Debate de Teses” (SDT) – descrito na Seção 2.2.1.

A partir das orientações da Linguagem AORML, elaboramos o DCU com a abstração dos agentes através de fronteiras. Decidimos utilizar essa linguagem de representação, pois ela permite que a utilização DCU seja feita conforme orientações da UML 2.0. Essa representação segue na Figura 6.

Figura 6 - Diagrama de Casos de Uso AORML - Sistema Multiagente Alpes



Fonte: Autoria Própria

Nas seções a seguir detalhamos os agentes e demais recursos que compõem o sistema.

5.1.1 Agentes

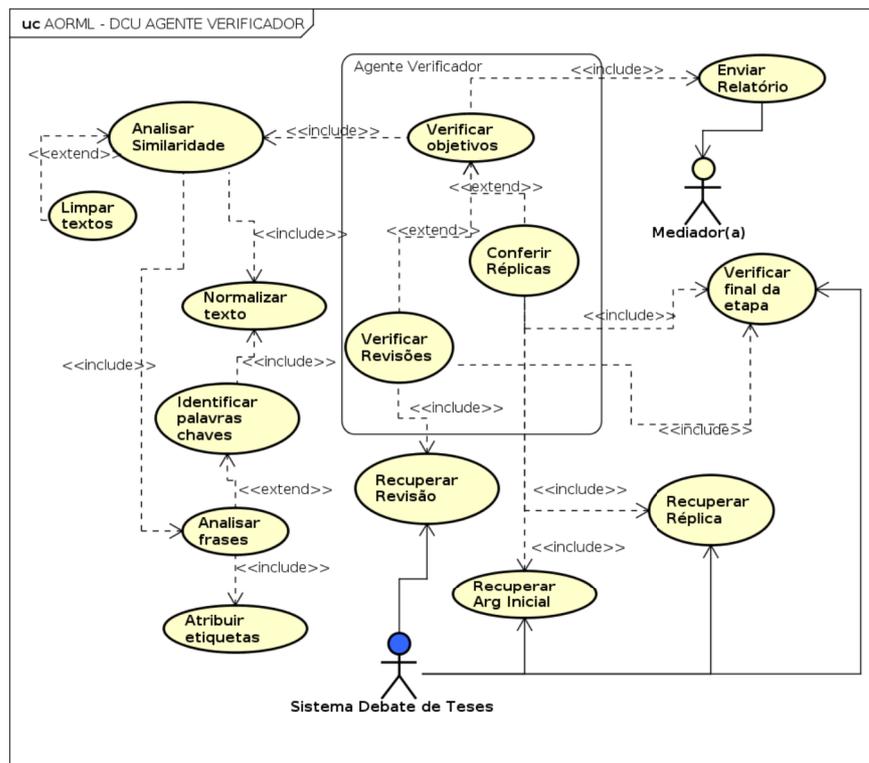
Como dito antes, os agentes do SMA ALPES são reativos, logo, a partir de alterações em suas áreas de influência os agentes são acionados. Apresentamos em sequência a descrição dos cinco agentes que compõe o SMA ALPES.

Nos DCU AORML de cada agente incluímos um nível maior de descrição dos casos de uso manipulados por cada agente. Como o SMA ALPES atua a partir das informações inseridas no SDT, este é representado nos diagramas como um ator. E os demais atores representados recebem as informações processadas pelos agentes.

5.1.1.1 Agente Verificador

O Agente Verificador, representado na Figura 7 por fronteiras, foi projetado para monitorar o envio das etapas de Revisão e Réplica, a fim de verificar se os textos produzidos nestas etapas estão de acordo com os objetivos da respectiva etapa.

Figura 7 - DCU AORML Agente Verificador



Fonte: Autoria Própria

Sendo assim, seu funcionamento inicia-se a partir do monitoramento do Calendário do Debate. Quando uma das condições “Término da Fase de Revisão” ou “Término da Fase de Réplica” é atendida, o agente inicia o fluxo de suas atividades.

- Condição: “Término da Fase de Revisão”

Na Etapa 3 – Revisão, o aluno revisor deve apresentar críticas e/ou sugestões sobre os argumentos apresentados pelo aluno revisado em sua argumentação inicial. Primeiro é necessário resgatar os textos produzidos pelo aluno nas fases de Argumentação Inicial e Revisão. De posse dos textos, o agente utiliza técnicas de análise de similaridade para verificar se a revisão possui semelhança com o texto da argumentação inicial, buscando por

elementos que estejam relacionados ao texto escrito na fase de Argumentação Inicial. Finalizada a análise dos textos, o agente envia uma notificação para o Mediador. A notificação é composta por um relatório que indica o valor da similaridade entre os textos analisados e os termos semelhantes são destacados para facilitar a leitura do mediador.

Além disso, o agente verifica se as duas revisões realizadas por um aluno são iguais, uma vez que um aluno é revisor de outros dois. Caso um aluno tenha realizados duas revisões iguais, o agente envia uma notificação para o Mediador, onde consta o nome do aluno e uma cópia dos textos idênticos.

Por fim, o agente busca por textos curtos como “Concordo com o que você disse” ou “Não acho que você está certo” que não contribuem para a discussão, uma vez que não apontam onde na argumentação inicial o aluno revisado pode melhorar. Por isso, com base em uma lista definida de textos curtos, buscamos esses padrões dentro das revisões, e caso sejam encontrados, o sistema alerta o professor para que ele comunique o aluno revisor e o oriente sobre como fazer a etapa de revisão.

- Condição: “Término da Fase de Réplica”

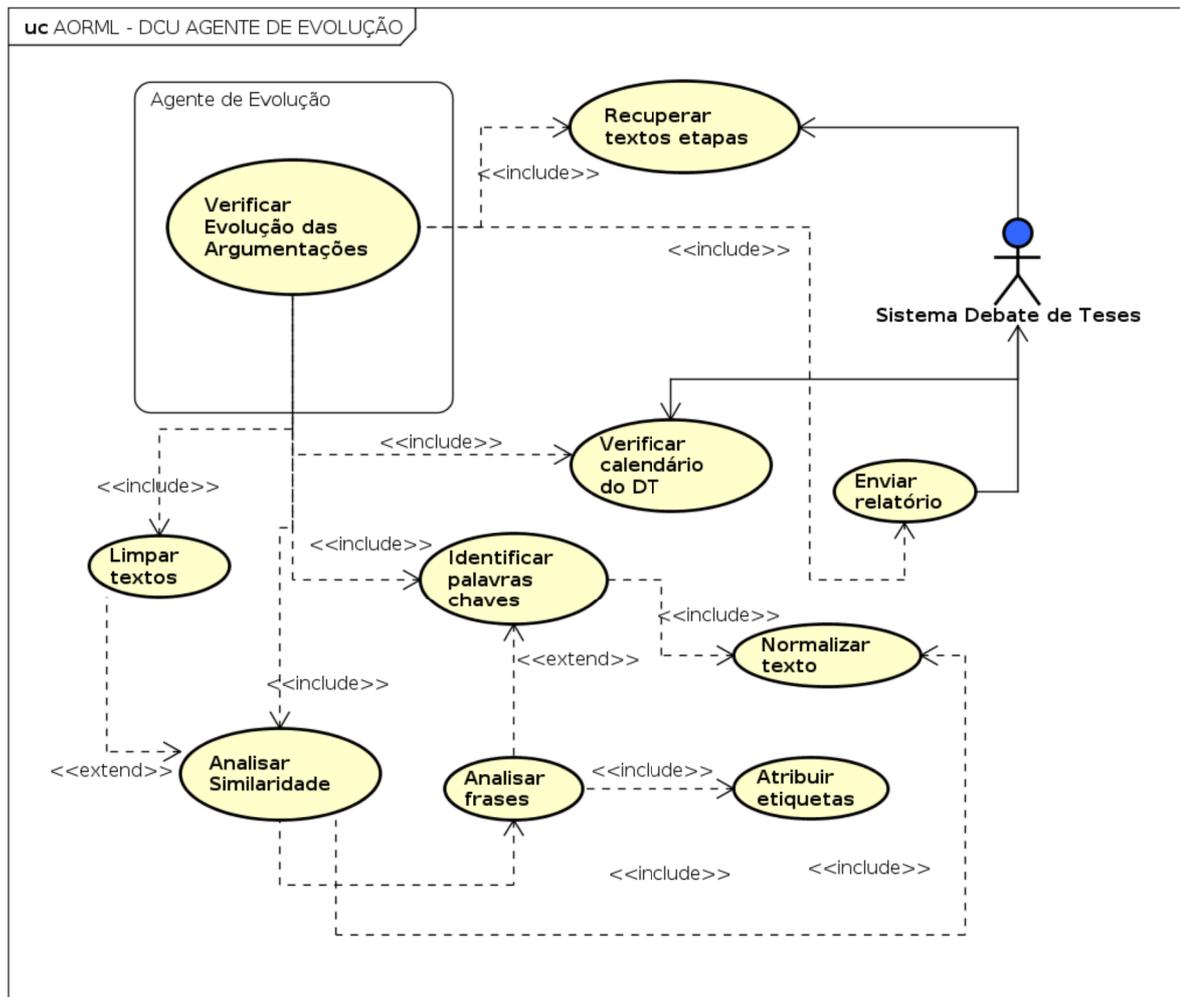
Os textos analisados são os produzidos nas fases de Revisão e Réplica. Diante disso, é necessário resgatá-los. O agente, de posse dos textos, aciona as funcionalidades que analisarão a semelhança entre os textos produzidos, destacar os termos semelhantes, e indicar se os textos produzidos nas réplicas são diferentes. Por fim, é enviado um relatório para o Mediador, com o valor de semelhança entre a Réplica e a Revisão. Caso sejam encontradas duas réplicas iguais, o agente envia uma notificação para o Mediador indicando o nome do aluno e com a cópia dos textos idênticos.

5.1.1.2 Agente de Evolução

A Figura 8 traz o DCU do Agente de Evolução. Este agente é responsável por comparar os textos produzidos na etapa de Argumentação Inicial com os demais textos das demais etapas, a busca de elementos que foram acrescentados na

Argumentação Final.

Figura 8 - DCU AORML Agente de Evolução



Fonte: Autoria Própria

Seu processo inicia-se ao final do debate, ou seja, quando o aluno tiver produzido sua argumentação final sobre a tese em discussão, e a data máxima para envio for ultrapassada em um dia.

Os textos produzidos pelo aluno durante o debate são recuperados do banco de dados do SDT. A primeira análise é realizada a partir da comparação entre os textos da Argumentação Inicial e Argumentação Final. Sobre os textos são aplicados processos para identificar quais são os novos elementos (substantivos, verbos, adjetivos) inseridos na argumentação final. Caso termos novos sejam identificados, estes termos são separados e servem de base para nova busca.

A nova busca consiste em verificar se os termos identificados estão fase Réplica.

Para isso, repete-se o processo de identificação de termos. Se todos os termos forem encontrados nesta fase da análise, o agente apresenta para o professor o texto da argumentação final com os novos termos destacados e o texto da réplica com os termos destacados.

Se sobraem termos não relacionados à Réplica, buscam-se esses termos nos textos produzidos na fase de Revisão. Caso todos os termos sejam encontrados, o agente apresenta para o professor o texto da argumentação final com os novos termos destacados e o texto da revisão com os termos destacados. Caso algum novo elemento tenha sido encontrado nas réplicas, o texto da réplica com os termos destacados também é apresentado.

Se sobraem termos que não foram relacionados às fases de revisão ou réplica, são analisados os textos produzidos pelos dois alunos que foram seus revisados. Se os termos novos forem encontrados nos textos produzidos pelos revisados na etapa de argumentação inicial, os termos são destacados. O agente apresenta para o professor o texto da argumentação final com os novos termos destacados, e o texto da argumentação inicial, de um dos alunos ou dos dois, com os termos destacados. Além dos textos das revisões e réplicas, com os termos destacados, caso tenham sido identificados em uma ou ambas etapas.

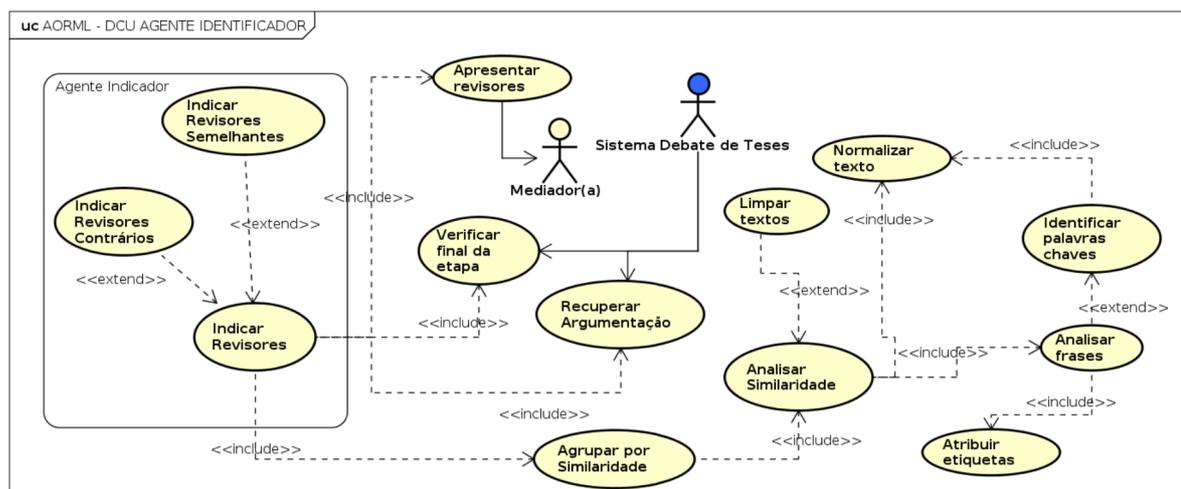
Caso ainda tenha algum termo que não foi relacionado, a última análise é realizada sobre o material didático enviado pelo professor para o sistema. Ao finalizar a busca pelos termos, o agente apresenta para o professor o texto da argumentação final com os novos termos destacados, o texto da argumentação inicial, de um dos alunos ou dos dois, com os termos destacados, o texto da réplica com os termos destacados também é apresentado, o texto da revisão com os termos destacados e os trechos do material didático aonde os termos foram encontrados.

O agente finaliza seu processo com a elaboração de um gráfico que apresenta a relação de influência de cada etapa na produção da argumentação final, com base na quantidade de termos novos da argumentação final e quantos destes termos foram encontrados nas buscas realizadas.

5.1.1.3 Agente Indicador

Na Figura 9 está o DCU do Agente Indicador. Este agente foi projetado para agilizar o processo de distribuição de revisores. Seu fluxo de ações é iniciado pelo Recurso Indicação de Revisores⁸.

Figura 9 - DCU AORML Agente Indicador



Fonte: Autoria Própria

Partindo da escolha do Mediador sobre qual será o critério inicial utilizado para distribuição dos revisores, recupera as informações sobre o posicionamento e argumentação inicial dos participantes.

Os critérios são “Indicar revisores com opiniões contrárias” ou “Indicar revisores com opiniões semelhantes”.

- Critério: “Indicar revisores com opiniões contrárias”

Inicia a análise pela divisão em dois grupos, um grupo formado pelos alunos que tenham indicado o posicionamento “Concordo” - Grupo A, e outro formado pelos alunos que tenham indicado o posicionado “Não concordo” - Grupo B. Compara a argumentação inicial de cada aluno do Grupo A com todos os alunos do Grupos B, verificando a semelhança entre os textos produzidos. Distribui como revisores de um aluno do Grupo A os dois alunos do Grupo B que tenham o menor valor semelhança encontrado ao comparar as argumentações.

⁸ Este e os demais recursos do SMA ALPES são explicados na Seção 5.1.2.

Caso o número de alunos do Grupo A seja maior que o número de alunos do Grupo B, e todos os alunos do Grupo B já tenham sido relacionados como revisores dos alunos do grupo A. Realiza a comparação das argumentações iniciais entre os alunos do Grupo A e atribui como revisores os dois alunos que tiverem as argumentações menos semelhantes. Essa lógica de distribuição serve também para o caso contrário, ou seja, o número de alunos do Grupo B é maior que o número de alunos do Grupo A.

- Critério “Indicar revisores com opiniões semelhantes”

Inicia a análise pela divisão em dois grupos, um grupo formado pelos alunos que tenham indicado o posicionamento “Concordo” - Grupo X, e outro formado pelos alunos que tenham indicado o posicionamento “Não concordo” - Grupo Y. Realiza a comparação das argumentações iniciais entre os alunos do grupo, atribuindo como revisores de um aluno, dois alunos que sejam membros de seu grupo e que possuem as argumentações iniciais mais semelhantes à do primeiro aluno. Caso o número de alunos de um grupo não seja suficiente para a atribuição de revisores, realiza a comparação das argumentações iniciais dos alunos do Grupo X que ainda não tiveram revisores atribuídos com as argumentações iniciais dos alunos do Grupo Y, e atribui como revisores os dois alunos do Grupo Y que tiverem as argumentações mais semelhantes e que não tiverem sido atribuídos como revisores de outros dois alunos.

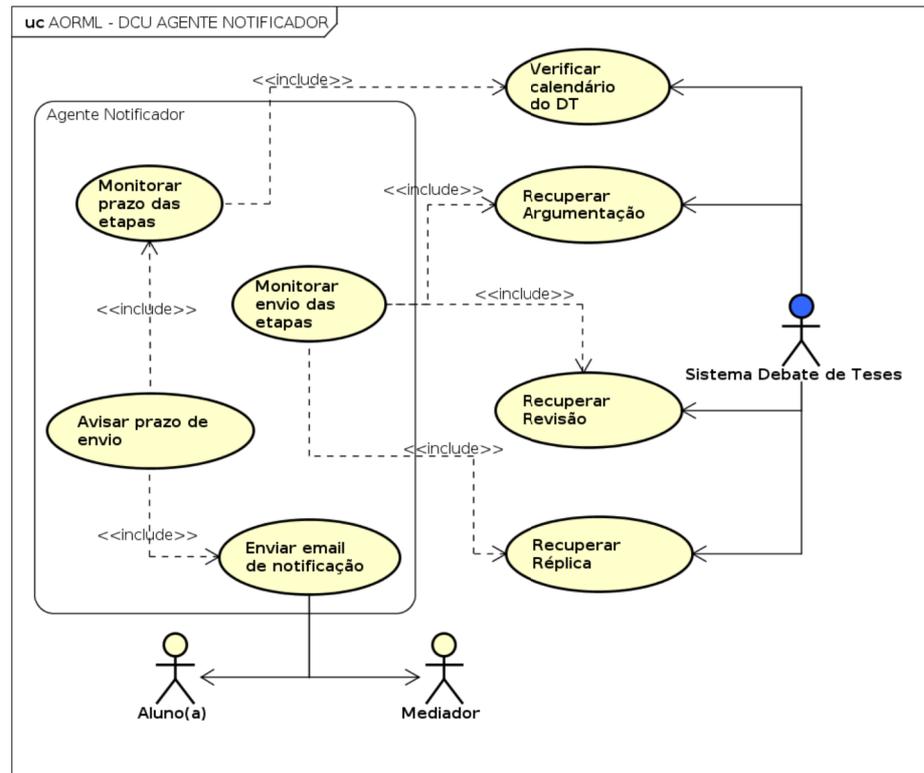
Nas duas distribuições, cuida-se para que não sejam criados subgrupos de revisão, ou seja, indicar o aluno A como revisor do aluno B e do aluno C, o aluno B como revisor do aluno A e C, e o Aluno C como revisor do aluno A e do aluno B. Além disso, atenta-se para o fato de que um aluno só pode ser revisor de outros dois.

Como resultado final, retorna para o Mediador a distribuição realizada para sua conferência e aceitação.

5.1.1.4 Agente Notificador

A Figura 10 mostra o DCU do Agente Notificador que é responsável por avisar os alunos que ainda não enviaram seus textos os prazos de cada etapa do debate, utilizando como base o calendário definido pelo mediador ao configurar o debate no SDT.

Figura 10 - DCU AORML Agente Notificador



Fonte: Autoria Própria

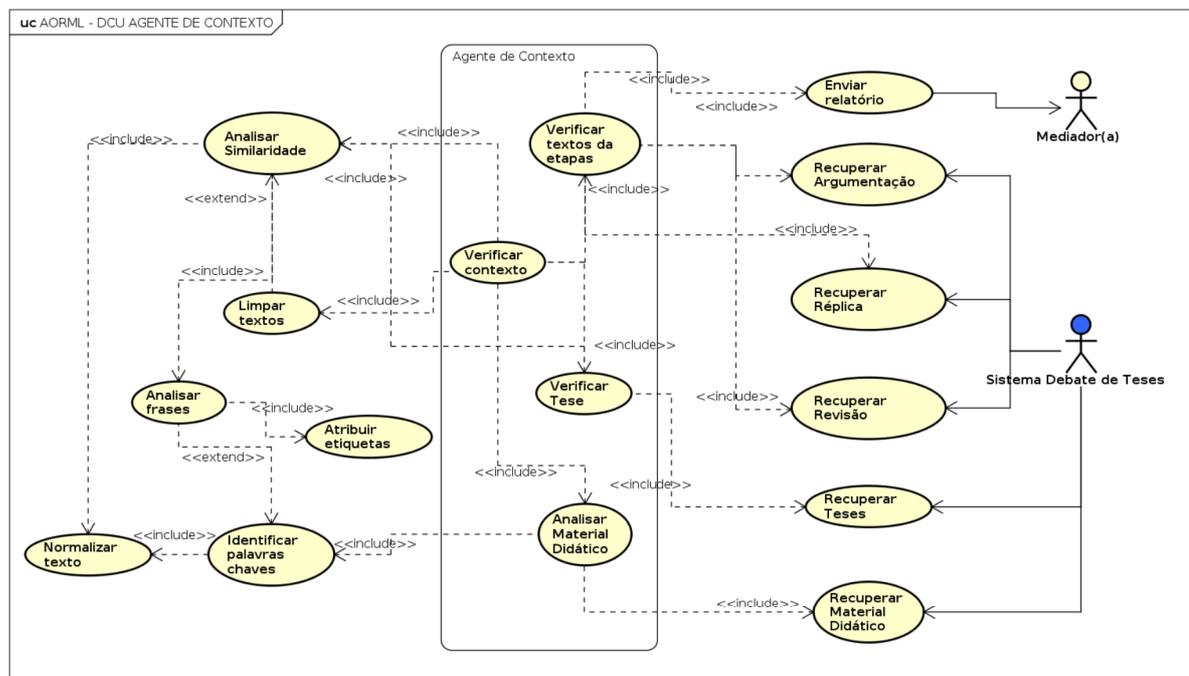
Assim, o Agente Notificador realiza o monitoramento dos envios das etapas, seguindo o cronograma do debate. Caso as etapas do debate tenham sido configuradas com intervalos de envio maiores que um (1) dia, este agente realiza uma sondagem diária enquanto o debate estiver ativo identificando quais alunos ainda não enviaram sua produção da etapa. Essa verificação é necessária para que sejam enviados lembretes por e-mail para os participantes do debate informando sobre a proximidade término da etapa. O mediador define a quantidade de dias de antecedência ao final de uma etapa que o agente deve considerar para enviar os e-mails. E também, o mediador configura a quantidade de vezes e dias de que ele deseja receber um relatório por e-mail com a listagem de alunos que ainda não

enviaram seus textos, e envia novamente um lembrete para os alunos. Ao finalizar o prazo de uma etapa, o agente envia para o Mediador uma lista de alunos que não produziram os textos na referida etapa.

5.1.1.5 Agente de Contexto

O Agente de Contexto - Figura 11 - inicia suas ações no momento em que um aluno realiza o salvamento ou envio do texto de qualquer uma etapa. Este agente monitora todas as etapas, a fim de auxiliar o aluno a produzir seus textos seguindo o tema proposto pela tese.

Figura 11 - DCU AORML Agente de Contexto



Fonte: Autoria Própria

Para isso, é necessário que o Mediador alimente o sistema com algum texto de referência (material didático, textos diversos sobre o assunto). Sobre os textos de referência são aplicadas técnicas para identificar quais as palavras-chaves sobre o assunto e para criação de uma base que será utilizada como um dicionário de relações para criar aproximações entre os textos produzidos pelos alunos e o tema em discussão. Nesta etapa, a tese proposta também é resgatada do banco de dados do SDT, ela também servirá de base para as análises.

Num segundo momento, o texto da etapa será analisado a partir do dicionário de relações e do texto da tese. Neste ponto, o agente tenta identificar quais elementos estão presentes no texto produzido pelo aluno, no dicionário e na tese. Caso não sejam encontrados elementos ou sejam encontrados poucos elementos que contenham essa relação, o Mediador recebe um relatório com o nome do aluno e o texto por ele produzido.

5.1.2 Recursos

A seguir explicamos cada recurso que está à disposição do mediador, elaborados a partir da análise das funcionalidades requeridas (Figura 5 e Figura 6).

5.1.2.1 Grupos de Similaridade

Este recurso está diretamente vinculado à mediação pedagógica “Agrupar indivíduos que possuem argumentações semelhantes”, e foi elaborado a partir do Caso de Uso “Criar grupos de Argumentação”.

Para criação dos Grupos de Similaridade, o professor deve indicar quais são os textos que serão analisados, escolhendo entre a Etapa 2 – Posicionamento e Argumentação Inicial e Etapa 5 – Posicionamento e Argumentação Final, e o número de grupos que devem ser formados.

A partir dessa escolha, os textos das etapas são resgatados e analisados. Sobre os textos são aplicadas técnicas de limpeza para exclusão de termos ou caracteres não importantes. Sobre os textos limpos são aplicadas técnicas para calcular a semelhança entre os textos e com base nessa semelhança criar os grupos de alunos. O resultado final é a criação de subgrupos de acordo com a semelhança entre os textos produzidos. Para o professor são apresentados os grupos formados contendo os nomes dos alunos de cada grupo.

5.1.2.2 Identificação de “similaridades” entre textos

Durante as interações do DT os alunos têm acesso aos textos produzidos por outros dois alunos – Etapas de Revisão. Por isso, para verificar se um aluno não se apropriou dos textos de outrem, analisamos se a Argumentação Final contém trechos semelhantes ao que foram escritos pelos outros alunos.

Para facilitar, utilizaremos o “AlunoA” como o aluno em análise. E “AlunoB” e “AlunoC” como alunos que foram revisados pelo “AlunoA”. Sendo assim, a argumentação final do AlunoA será comparada, com o uso de técnicas para calcular a semelhança entre textos, com a argumentação inicial do AlunoB e do AlunoC.

Caso, na argumentação final do AlunoA sejam encontrados trechos totalmente semelhantes aos das argumentações iniciais do AlunoB e do AlunoC, o sistema retornar para o professor os textos produzidos em todas as etapas do debate pelos AlunoA, AlunoB e AlunoC. Dessa forma, o professor poderá realizar uma análise mais direcionada para o caso, orientando que o AlunoA faça referência aos textos incorporados à sua argumentação final.

5.1.2.3 Indicação de Revisores

Como dito na descrição do Agente Indicador, esse recurso inicia o funcionamento do referido agente no sistema, pois é necessário que um critério seja atendido. Esse critério está relacionado a escolha do tipo de distribuição que será realizada.

Para isso, o professor deve escolher entre (a) indicar alunos com opiniões contrárias ou (b) indicar alunos com opiniões semelhantes. Caso o professor tenha escolhido a opção (a) são indicados como revisores alunos que possuem posicionamentos contrários e uma semelhança baixa entre as argumentações. Mas, se o professor tiver escolhido a opção (b) são indicados como revisores alunos que possuem o mesmo posicionamento e uma semelhança média entre as argumentações.

Destacamos que a escolha da opção (a) pode ser usada para promover ao aluno acesso a pontos de vistas diferentes sobre a mesma tese. Dessa forma, os revisores terão que buscar por novos conhecimentos para verificar se a argumentação revisada está coerente e possui argumentos válidos.

Como resultado, o Agente Indicador manipula a página de delegação de revisores dentro do SDT, atribuindo para cada aluno seu par de revisores. Ao professor cabe verificar se a distribuição está de acordo com o desejado e aceitar (ou não) a distribuição.

5.1.2.4 Análise de Cordialidade

Este recurso está relacionado a mediação “Verificar se houve quebra de cordialidade nas revisões e réplicas”. Seu funcionamento depende de um pequeno dicionário composto por termos impróprios simples ou compostos e pequenas frases consideradas agressivas, previamente cadastradas no banco de dados da solução.

A partir da aplicação de técnicas para analisar a semelhança entre palavras, verifica-se se os termos cadastrados estão incluídos dentro dos textos produzidos nas etapas de Revisão e Réplica. Caso seja encontrado algum termo ou frase, o sistema apresenta para o professor o nome do aluno e o texto por ele produzido.

5.1.2.5 Identificação de Polaridade

Este recurso serve para verificar se o aluno fundamentou seu argumento de acordo com o posicionamento indicado. Ou seja, se o aluno indicou que “Concorda” com a tese em análise sua argumentação deve ser favorável à tese e possuir fundamentos que sustentem seu posicionamento.

Para isso verificamos se determinadas palavras-chave estão escritas em sua argumentação e se possuem relação com o “sentido” do posicionamento. Essas palavras-chave podem ser definidas pelo professor e adicionadas a uma lista de termos já existente no BD da solução.

5.1.2.6 Síntese das Autorias

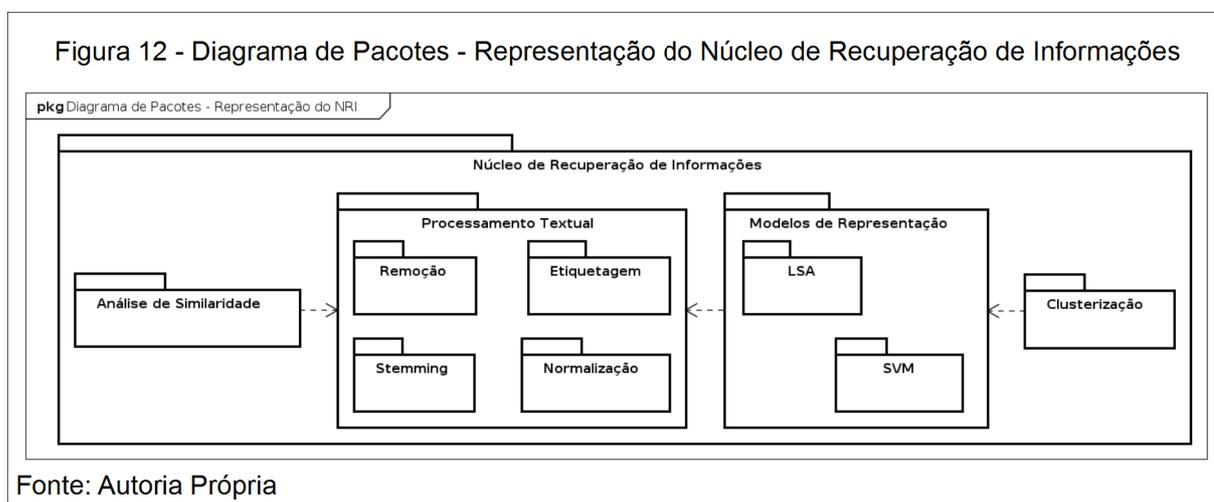
Para facilitar a elaboração de *feedback* para um determinado grupo de alunos, utilizamos técnicas para identificar quais são os termos mais frequentes entre as argumentações finais dos alunos que pertençam a um mesmo grupo. Os grupos são definidos com a aplicação dos mesmos processos utilizados no recurso Grupos de Argumentação.

Apresentamos para o professor os termos de maior destaque na forma de Nuvem de Palavras. Assim, o professor poderá visualizar quais os termos mais frequentes entre as argumentações finais de alunos que fazem parte de um mesmo grupo, e a partir disso elaborar seu *feedback*.

5.2 NÚCLEO DE RECUPERAÇÃO DE INFORMAÇÕES

O Núcleo de Recuperação de Informações (NRI) foi elaborado usando técnicas, métodos e algoritmos apresentados nos Capítulos 3 e 4. Portanto, o NRI foi construído como um conjunto de pacotes, onde cada pacote contém classes que são responsáveis por analisar e processar os textos a fim de recuperar informações relevantes, dada uma determinada solicitação (entrada).

Na Figura 12 mostramos o Diagrama de Pacotes do NRI. Nela temos os quatro pacotes internos: Processamento Textual, Modelos de Representação, Análise de Similaridade, Clusterização. O pacote Processamento Textual possui quatro pacotes internos: Remoções, Etiquetagem, Stemming e Normalização. O pacote Modelos de Representação contém dois pacotes internos, LSA e SVM. Além disso, estão exemplificadas as relações de dependência entre os pacotes.

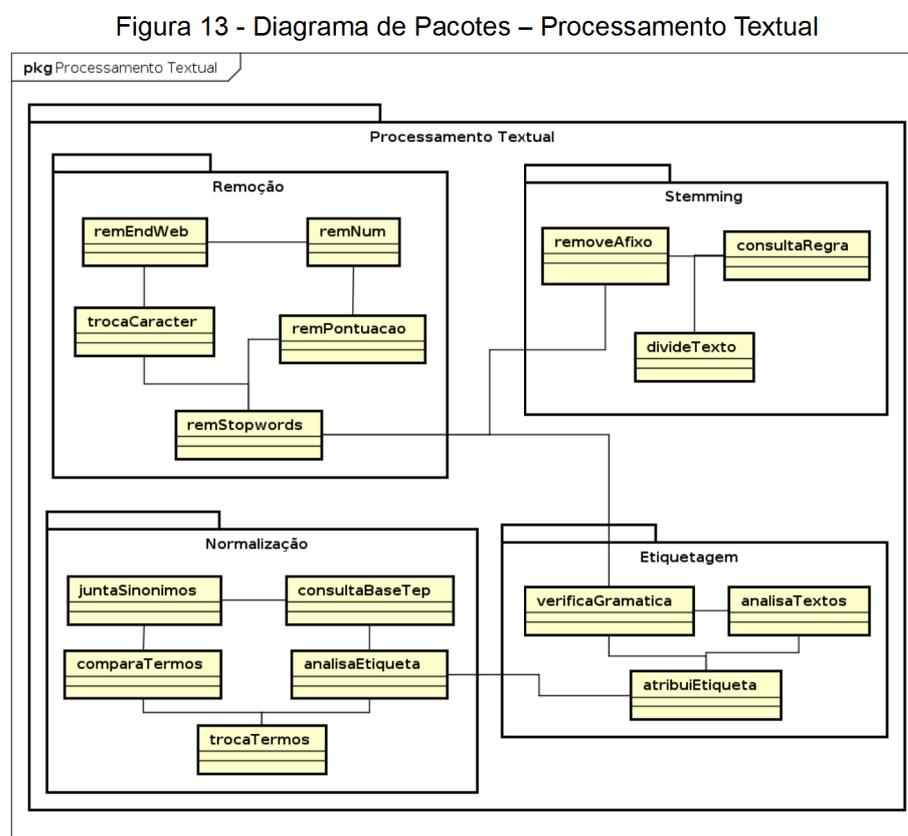


A seguir descrevemos a criação de cada pacote, iniciando pelo pacote de Processamento Textual.

5.2.1 Processamento textual

O pacote mais importante do Núcleo de Recuperação de Informações, é composto por processos responsáveis pela preparação do texto para que a recuperação de informações seja otimizada. Os processos foram selecionados de forma que, ao final de sua aplicação tenhamos um *corpus* de análise que será utilizado como entrada pelos demais processos.

Na Figura 13 apresentamos o Diagrama de Pacotes do Processamento Textual, com a descrição de seus pacotes internos com suas respectivas classes.



Fonte: Autoria Própria

O pacote **Remoção** contém classes que são responsáveis por técnicas de limpeza, ou seja, técnicas responsáveis por excluir do texto em análise caracteres ou palavras considerados irrelevantes para o processo de análise. As técnicas são descritas abaixo:

- Remoção de endereços *web*: Possíveis endereços *web* inseridos como referências para acesso a uma página *web* são removidos.

- b) Remoção de números: Esta tarefa consiste em remover do texto algarismos numéricos.
- c) Capitalização: Para facilitar a análise, trocamos todos os caracteres que compõem o texto pelos seus correspondentes em minúsculo.
- d) Remoção de pontuação: Esta tarefa consiste em tratar o texto removendo pontuações como vírgulas, pontos finais, travessões, etc.
- e) Remoção de Termos Frequentes ou Remoção de *Stopwords*: Este processo foi desenvolvido de duas maneiras. Na primeira, com base numa lista de palavras, buscamos no texto em análise as palavras da lista e as retiramos do texto. A segunda maneira consiste na análise da identificação sintática da palavra, sendo necessário primeiro que o texto seja analisado e etiquetado por um etiquetador morfossintático. Mantemos no texto apenas palavras etiquetadas como substantivos, verbos ou adjetivos. Os demais termos são descartados por serem utilizados apenas para auxiliar na leitura humana, não sendo importante para a análise computacional.

O pacote **Etiquetagem** contém as classes responsáveis pela implementação do Etiquetador Morfossintático, que consiste na análise das palavras de uma frase e sua respectiva classificação sintática.

No contexto desta pesquisa, desenvolvemos uma versão do Etiquetador Morfossintático (*POS Tagger*) proposto por Fonseca e Rosa (2013) no projeto NPLNet. Além disso, as etiquetas seguem o padrão proposto no *corpus* Mac-Morpho de Aluísio e outros (2013). Os detalhes sobre o projeto NPLNet e sobre o *corpus* Mac-Morpho foram descritos no Capítulo 3.

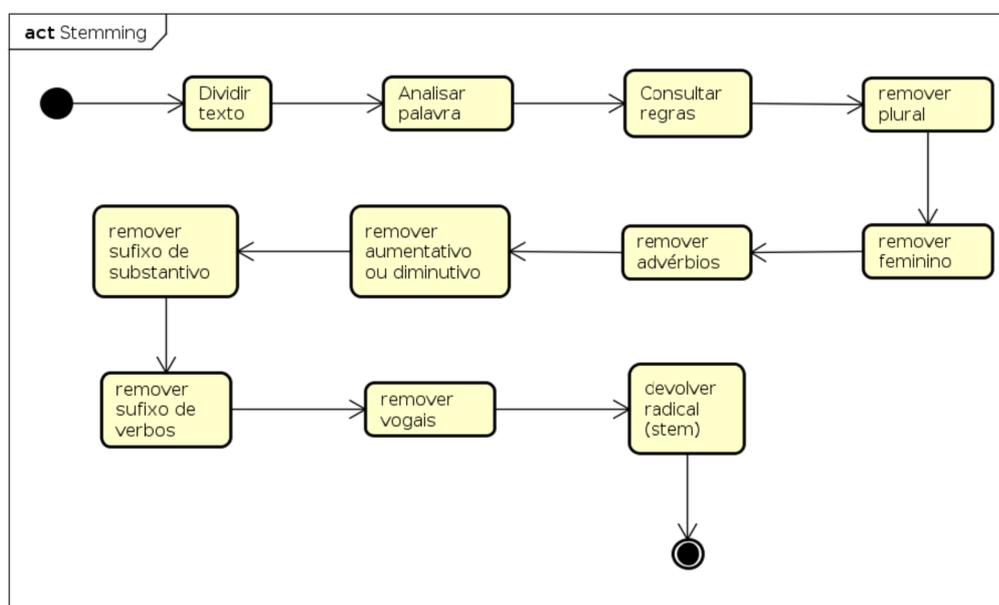
O etiquetador automático analisa cada palavra que compõe o texto, seguindo as regras de formação de frases do idioma Português, a fim de atribuir a palavra à etiqueta relacionada à sua classe gramatical. Ao final da aplicação do etiquetador, as palavras contidas no texto são representadas por uma tupla de dados com a Codificação Unicode⁹. No diagrama representamos um fragmento de texto original e o mesmo texto após ter sido processado pelo etiquetador, através das notas.

⁹ Unicode é um padrão que permite aos computadores representar e manipular, de forma consistente, texto de qualquer sistema de escrita existente. Este padrão não representa caracteres acentuados, e os substituí por sua referência dentro da tabela de codificação.

O pacote **Stemming** contém as classes responsáveis pela aplicação da técnica para redução de uma palavra ao seu radical (*stem*) de formação.

O processo de *Stemming* foi desenvolvido com base no Remove de Sufixo para Língua Portuguesa (RSPL) proposto do Orengo e Huyck (2001). O RSPL é composto por um conjunto de regras. Cada regra foi definida seguindo as regras de formação de palavras do idioma Português. O RSPL foi desenvolvido para tratar as particularidades do idioma Português. Diante disso, sua maior complexidade está na elaboração das regras e na especificação das exceções a cada regra existente. A Figura 14 mostra o fluxo de atividades da técnica de *Stemming*.

Figura 14 - Fluxo de atividades da técnica *Stemming*



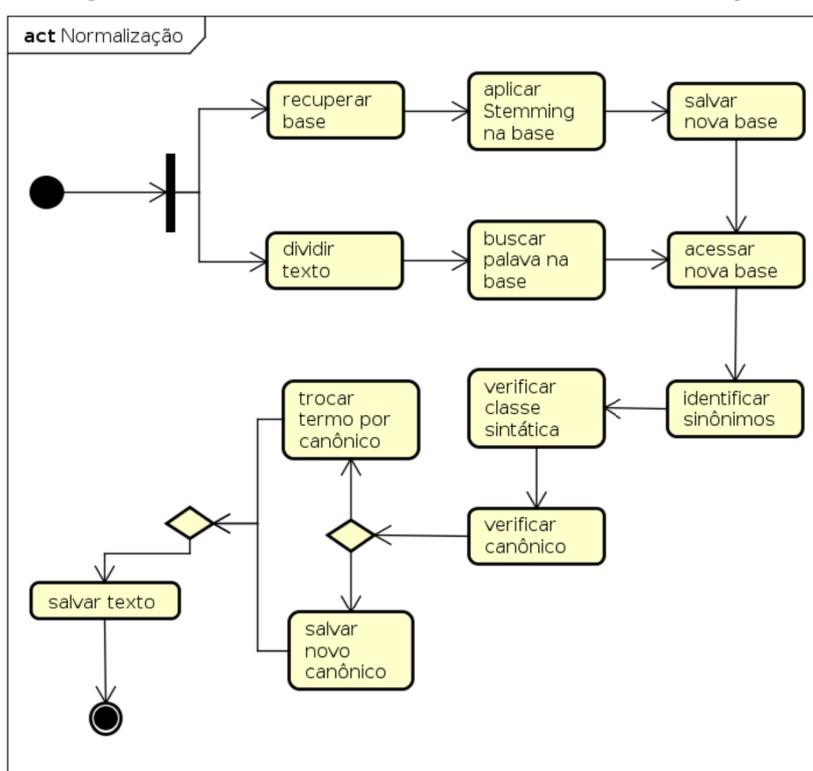
Fonte: Autoria Própria

Logo, cada palavra é analisada seguindo o conjunto de regras definidas. A primeira análise consiste na verificação de plural, caso a palavra analisada esteja no plural, esses caracteres são excluídos. A segunda análise é a verificação de advérbio, caso seja identificada, os caracteres também são removidos. Em seguida é analisado se a palavra está no feminino, se sim os caracteres são excluídos. A quinta análise serve para verificar se a palavra está no aumentativo ou no diminutivo. Depois é feita a redução de sufixo do substantivo, e segue para a redução de sufixo de verbos. Por fim, faz-se a redução de vogal. O resultado é o retorno apenas do radical (*stem*) de formação da palavra.

O pacote **Normalização** contém as classes responsáveis pela análise da relação de sinonímia entre as palavras do texto com uma lista de sinônimos. Esse pacote e suas respectivas classes representam o método de Normalização.

A Normalização consiste na troca de termos sinônimos por uma única forma de representação dentro do *corpus*, ou seja, termos sinônimos passam a ser representados por um único termo, denominado termo canônico. A Figura 15 mostra o fluxo de atividades do método de Normalização.

Figura 15 - Fluxo de Atividades do Método de Normalização



Fonte: Autoria Própria

Esse método foi desenvolvido com base no arquivo da WordNet.Br disponibilizado pelo projeto TeP2.0¹⁰. O arquivo possui 19885 entradas com as relações de sinonímia da língua portuguesa para verbos, substantivos, adjetivos e advérbios. Abaixo apresentamos o formato das entradas no arquivo.

```

NUM1. [Tipo] {termos sinônimos} <NUM2>
NUM1 = NUMERO DA LINHA DE REFERÊNCIA PARA TERMO SINÔNIMO
NUM2 = NUMERO DA LINHA DE REFERÊNCIA PARA TERMO ANTÔNIMO
263. [Verbo] {consentir, deixar, permitir} <973>
  
```

10 Mais informações em <http://143.107.183.175:21480/tep2/index.htm>. Acesso em 15.01.2016

Diante das peculiaridades do nosso *corpus* de análise, optamos por utilizar como entrada para o método de normalização o *corpus* resultante da aplicação da técnica de *Stemming*. Uma vez que todas as palavras que compõem o *corpus* foram reduzidas ao seu radical de formação aumentamos a chance de encontrarmos mais sinônimos para um termo.

Seguindo o fluxo de atividades apresentado na Figura 15, para realizarmos a busca na base TeP2.0, foi necessário a aplicação da técnica de *Stemming* sobre a base. Outro ponto é a definição do termo canônico, para isso utilizamos a primeira identificação de sinonímia encontrada para um conjunto de sinônimos. Além disso, para verificarmos a relação de sinonímia, primeiro analisamos qual a classe sintática a que palavra pertence, ou seja, se a palavra tiver sido classificada como verbo pelo etiquetador morfossintático, analisaremos apenas as relações de sinonímia com verbos. Mesmo que o radical de formação seja o mesmo, priorizamos a classe gramatical para buscar os possíveis sinônimos.

O *corpus* resultante do processo de normalização consiste em textos compostos por termos canônicos, assim todos os termos sinônimos são representados pelo mesmo termo, o que auxiliará a Análise de Similaridade.

5.2.2 Modelos de Representação

Este pacote é composto por dois modelos de representação, representados pelos pacotes LSA e SVM. O pacote LSA, detalhado na seção 5.2.3.1, é composto por classes responsáveis pela implementação da técnica de Análise Semântica Latente. O pacote SVM, detalhado na seção 5.2.3.2, contém as classes responsáveis pela implementação do Modelo Espaço-Vetorial e da métrica TF-IDF.

5.2.2.1 Módulo LSA

Como explicamos no Capítulo 3, a LSA foi criada para tratar grandes bases de documentos, e serve para extrair e representar o significado semântico de palavras em um contexto, obtidos através de cálculos estatísticos aplicados a um conjunto numeroso de textos. Seu modelo de indexação semântica é baseado na

coocorrência de palavras em textos. Contudo, encontramos na literatura científica indicativos que sua aplicação a pequenas bases gera bons resultados, conforme detalhamos no Capítulo 4. Contudo, em termos práticos, a LSA é utilizada para criar uma representação semântica, onde os textos podem ser representados por vetores. Para seu melhor aproveitamento, optamos por aplicar a remoção de pontuação e remoção de *stopwords* nos textos que serão analisados.

O espaço semântico criado pela LSA precisa ser treinado, ou seja, é necessário alimentar o sistema com textos que contenham as relações semânticas que gostaríamos de identificar. Dessa forma, são criados dicionários que posteriormente servirão de base para auxiliar na análise de similaridade entre os textos. Diante disso, utilizamos duas abordagens para criação dos espaços semânticos. A primeira utiliza o texto produzido dentro do AVEA, e a segunda utiliza textos utilizados como materiais de referência ou didáticos.

O resultado final de aplicação da LSA, é uma matriz de representação da frequência e importância das palavras dentro de um *corpus* e que considera a proximidade entre duas palavras e sua possível relação semântica.

5.2.2.2 Modelo Espaço-Vetorial – SVM

No SVM – Modelo Espaço-Vetorial os documentos são representados como vetores e as dimensões são representadas pela frequência das palavras que compõe os documentos (SALTON, WONG e YANG, 1975).

A frequência é calculada pela medida TF-IDF. Desta forma, a importância de um termo é relativa a sua frequência em cada documento da coleção, e inversamente proporcional ao número de documentos em que a palavra aparece.

5.2.3 Análise de Similaridade

O pacote Análise de Similaridade contém a classe que representa a implementação da métrica Similaridade de Cossenos, e a utiliza como base para calcular a similaridade entre textos e verificar quão semelhantes dois ou mais textos são.

Diante desse cenário, a Análise de Similaridade foi desenvolvida com base na métrica Similaridade de Cossenos. Essa é a métrica mais recomendada para análise de textos de acordo com Feldman e Sanger (2007), como *corpus* de análise é composto por textos muito pequenos, calcular a similaridade entre eles torna-se uma tarefa complexa, por isso optamos pela métrica recomendada.

5.2.4 Clusterização com K-Means

O pacote Clusterização foi elaborado com base na classe que representa a implementação no algoritmo K-Means. Para sua aplicação é necessário que o *corpus* esteja representado seguindo um modelo computacional. Para isso, utilizamos duas abordagens para sua utilização. A primeira utiliza como modelo de análise as matrizes resultantes da aplicação da LSA. A segunda utiliza o espaço-vetorial formado com a aplicação de SVM.

O próximo passo segue na partição do *corpus* e na escolha aleatória de um número determinado de centróides. Os centróides são vetores de documentos que serão utilizados como base para calcular a semelhança entre eles e os demais documentos que compõem o *corpus*. O número de centróides é igual ao número de grupos que devem ser formados.

O K-Means inicia sua análise de similaridade com a aplicação da Distância Euclidiana. Essa métrica calcula distância entre dois vetores de documento no espaço. Assim, quanto mais próximo de zero for o valor final da distância, mais semelhantes são os documentos. Para pertencer a um mesmo grupo, é necessário que o documento possua semelhança com todos os candidatos a membro deste grupo. Como condição de parada, o K-Means realiza n iterações, sendo que n é o número de iterações necessárias para estabilização dos membros em um grupo. Este número pode ser pré-definido ou o algoritmo pode ser implementado de forma a controlar essa estabilização. Para definir o número de iterações necessárias para criação dos grupos, é necessário conhecer os textos que constituem o *corpus*.

5.3 CONSIDERAÇÕES FINAIS DO CAPÍTULO

A solução computacional descrita neste capítulo foi concebida para facilitar que o Mediador conduza mediações pedagógicas durante um “Debate de Teses”. Para isso, identificamos as necessidades (Figura 5) que poderiam ser modeladas computacionalmente. A partir da análise dessas necessidades, optamos por elaborar uma solução baseada na Arquitetura de Sistema Multiagente (Figura 6), pois identificamos que algumas tarefas poderiam ser automatizadas. A essa solução damos o nome de Sistema Multiagente ALPES.

Para subsidiar as ações dos agentes e dos demais recursos do sistema, criamos o Núcleo de Recuperação de Informações (NRI). Objetivo computacional principal da pesquisa, o NRI foi concebido a partir dos métodos, técnicas e algoritmos apresentados na Seção 3.2, que são utilizados pelos agentes e demais recursos dos sistemas para alcançarem seus objetivos.

Assim, para comprovar que os recursos disponibilizados pelo Sistema Multiagente ALPES mostram-se como uma solução favorável para apoiar o Mediador, sistematizamos uma prova de conceitos que segue descrita no Capítulo 6.

6 PROVA DE CONCEITO

Neste capítulo apresentamos a Prova de Conceito da solução computacional proposta no Capítulo 5, Sistema Multiagente Alpes. Para isso, na Seção 6.1 apresentamos os detalhes da análise e desenvolvimento do protótipo computacional. Na Seção 6.2 mostramos com detalhes os recursos oferecidos nesta versão do protótipo. E, na Seção 6.3 trazemos as considerações do capítulo.

Na condução deste estudo foram implementadas todas as funcionalidades do Núcleo de Recuperação de Informações (NRI) e os recursos Grupos de Similaridade e Síntese das Autorias. O recurso Grupos de Similaridade foi escolhido pois para sua perfeita execução é necessário que todas as funcionalidades do NRI também fossem implementadas. Além disso, a criação manual de grupos de alunos mostrou-se como uma das tarefas mais custosas em relação ao tempo do mediador, o que nos motivou a comparar os resultados alcançados pela solução aos resultados obtidos de forma manual. Já o recurso Síntese das Autorias foi escolhido por utilizar parte do NRI para sua execução e porque essa atividade necessita que o mediador leia várias vezes os textos dos alunos para conseguir destacar os termos que ele considere como mais importantes.

6.1 PROTÓTIPO COMPUTACIONAL

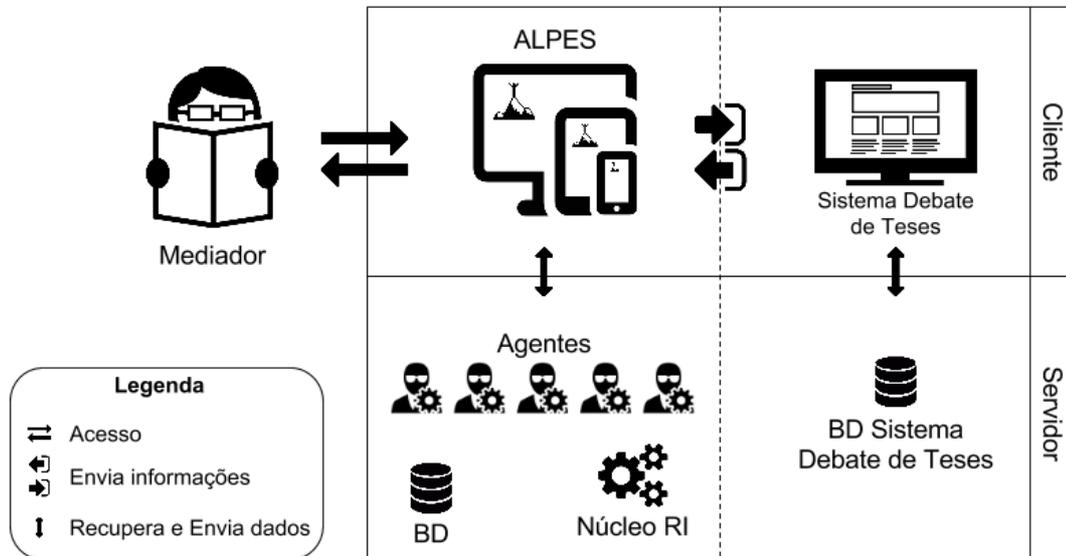
Com o objetivo de validar a solução proposta no Capítulo 5, em relação a recuperação de informações em textos, desenvolvemos um protótipo computacional do Núcleo de Recuperação de Informações (Seção 5.2). A partir do núcleo o sistema disponibiliza os recursos descritos na Seção 5.1.2.

O protótipo é um sistema *web*, desenvolvido com base no Framework Django 1.6 (DJANGO, 2014) e na linguagem de programação *Python 2.7.6* (PYTHON, 2014). E, para garantirmos que o sistema pode ser acessado em dispositivos diversos que possuam um navegador *web* e acesso à internet, a interface de acesso foram desenvolvidas a partir de modelos disponibilizadas pelo Projeto Bootstrap¹¹. Por ser

¹¹ Mais informações em <http://getbootstrap.com/about/>

um sistema *web*, o desenvolvimento do protótipo segue a Arquitetura Cliente-Servidor. Na Figura 16 apresentamos uma abstração do protótipo seguindo esse modelo de arquitetura.

Figura 16 - Arquitetura Cliente-Servidor do Protótipo

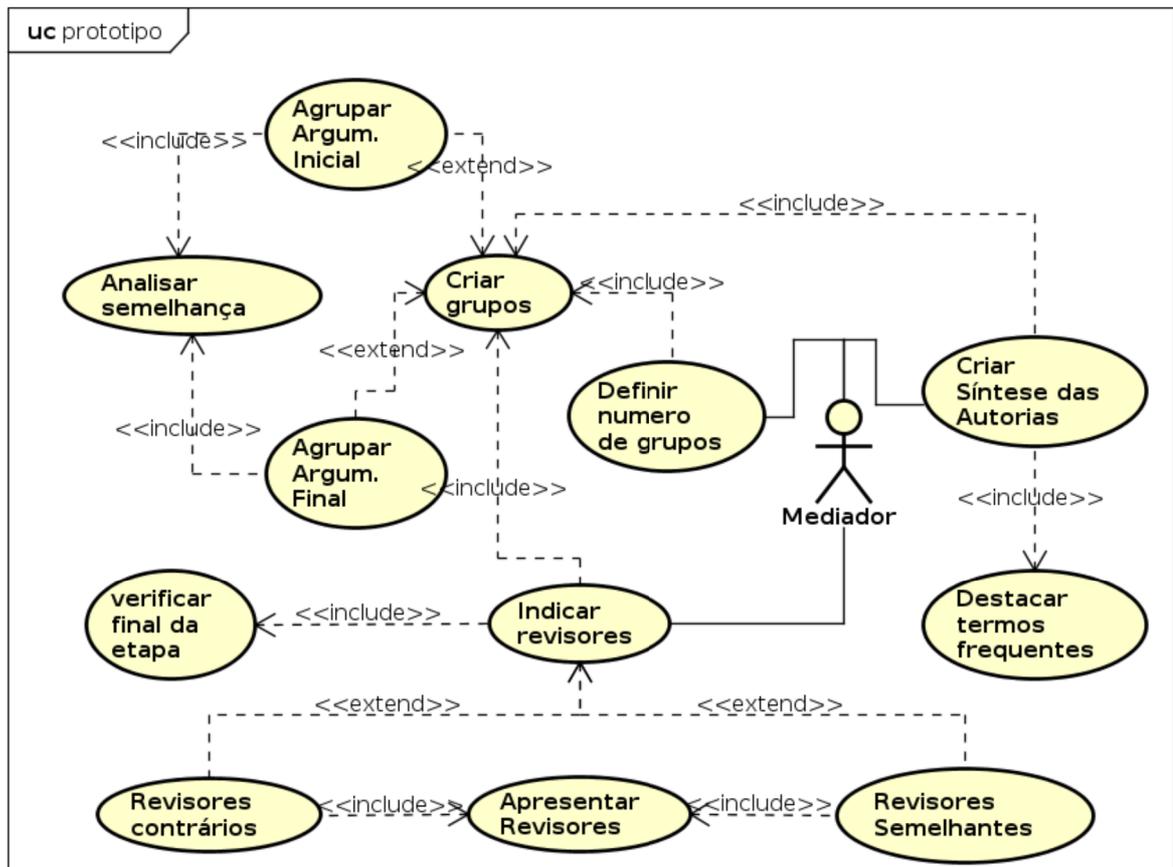


Fonte: Autoria Própria

Na camada Cliente (Figura 16) temos a interação de acesso entre Mediator e o Alpes, o envio de informações entre o Alpes e o Sistema “Debate de Teses” (SDT). Na camada Servidor temos os componentes de ambos sistemas, sendo que os componentes do Alpes são seu banco de dados, o Núcleo de Recuperação de Informações (NRI) e os Agentes. Já o SDT tem seu banco de dados.

Na fase de análise do protótipo destacamos os Casos de Uso para sua construção. Os Casos de Uso apresentados na Figura 17 contemplam as funcionalidades requeridas pelos recursos disponibilizados nesta versão protótipo.

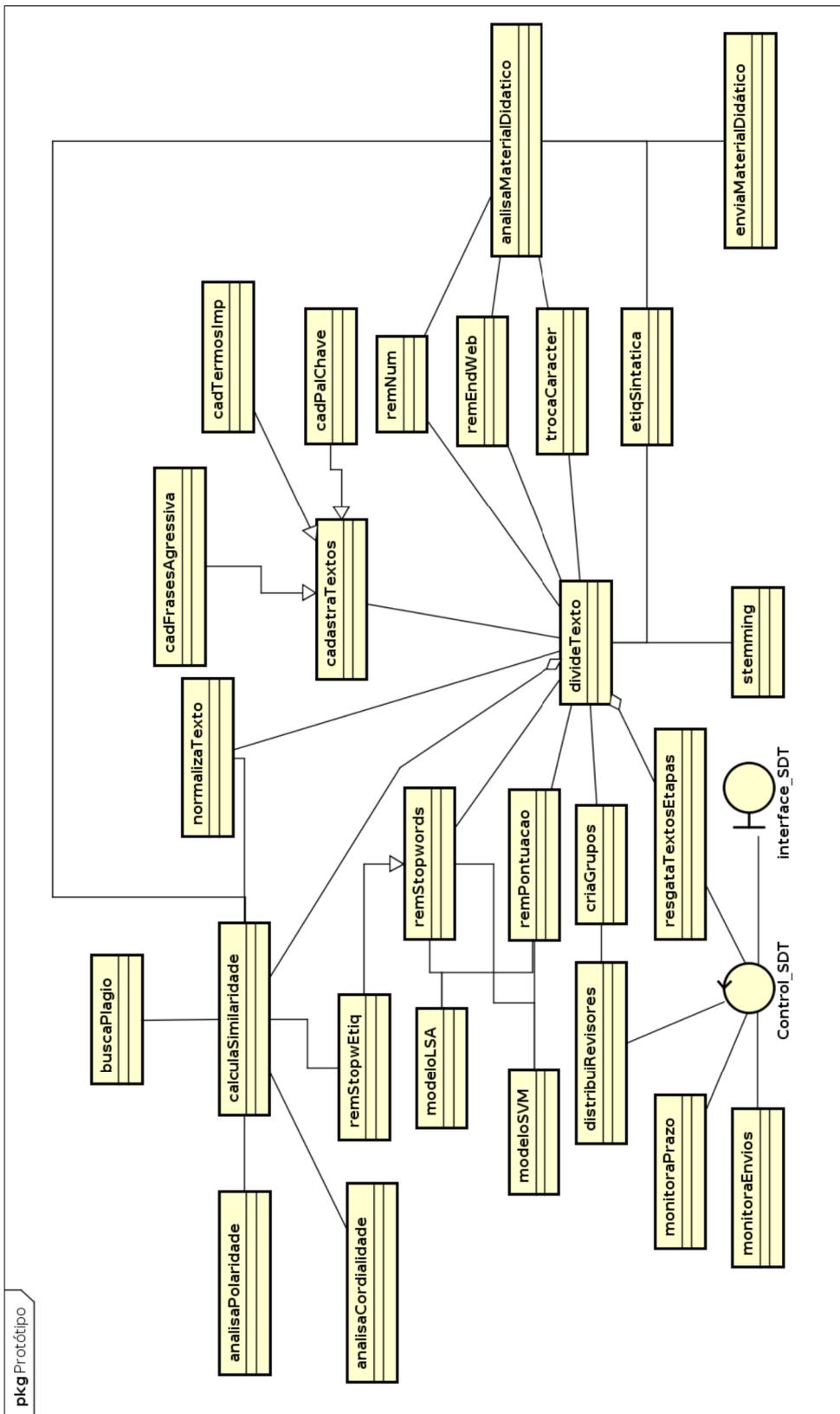
Figura 17 - Diagrama de Casos de Uso do Protótipo



Fonte: Autoria Própria

A partir da análise dos casos de uso, elaboramos o diagrama de classes do protótipo, apresentado na Figura 18. Salientamos que as classes que compõem o protótipo representam a criação completa do Núcleo de Recuperação de Informações.

Figura 18: Diagrama de Classes do Protótipo

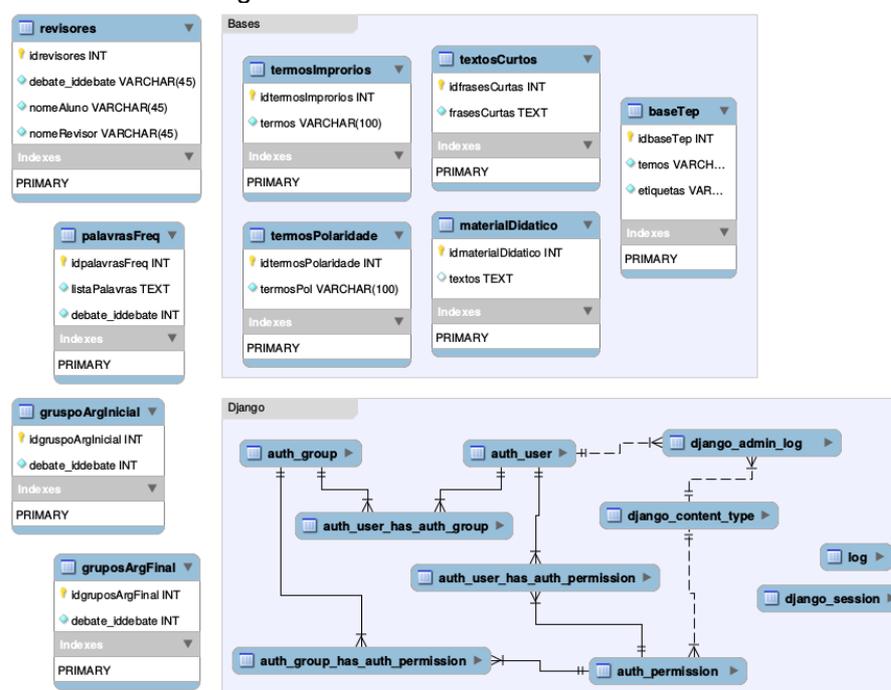


Autoria Própria

Na Figura 19, apresentamos o Diagrama Entidade-Relacionamento do BD Solução. Nele estão representadas as tabelas que compõem o banco de dados do Alpes, com ênfase para as tabelas destacadas como **Bases** que armazenam informações que são enviadas pelo mediador e que posteriormente são utilizadas como base de comparação para algumas ferramentas. E também a tabela *baseTep* que populamos com as informações contidas no arquivo disponibilizado pelo projeto.

Por padrão, o *Django* cria algumas tabelas no BD para gerenciar funcionalidades importantes para este tipo de sistema, como gerenciamento de usuário e controle de sessões, conforme destaque.

Figura 19 - DER Banco de dados ALPES



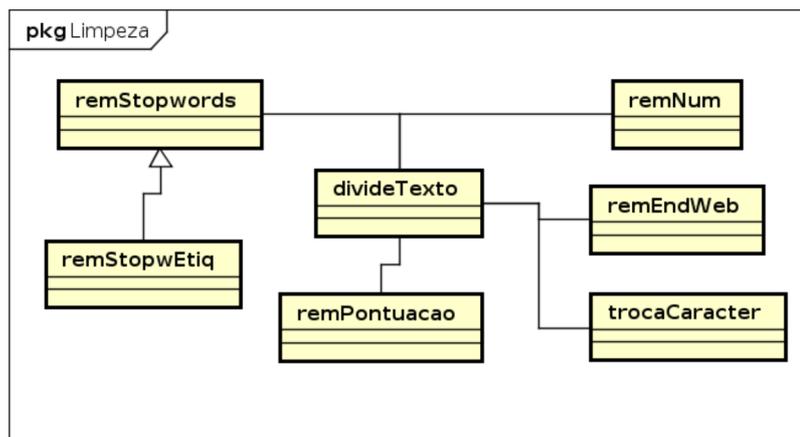
Fonte: Autoria Própria

Conseqüentemente, a seguir retratamos o desenvolvimento das técnicas e algoritmos utilizados na construção das classes descritas na Figura 18.

6.1.1 Classes responsáveis pela limpeza do texto

As classes que estão representadas na Figura 20 são responsáveis pela remoção de caracteres ou termos irrelevantes do texto que será analisado.

Figura 20 - Recorte do DC Protótipo - Classes responsáveis pela limpeza do texto



Fonte: Autoria Própria

A implementação de cada classe é descrita a seguir.

- Classe remEndWeb: Classe desenvolvida para retirar do texto todas as referências realizadas na forma de *links* para endereços *web*. Essa classe foi desenvolvida com o uso da biblioteca de expressões regulares re, a partir da definição das expressões regulares [r"(?:\@|https?\:\/\/)\S+"] e [r"(?:\@|www?\.)\S+]. Dessa forma, ao analisar um texto, busca-se por este padrão de escrita, caso seja encontrado, todo o *link* é removido do texto.
- Classe remNum: Essa classe foi implementada para analisar os textos à procura de caracteres numéricos. A partir da função replace da biblioteca de Strings, analisamos as palavras que compõem o texto, excluindo todas as referências numéricas encontradas por espaços em branco.
- Classe trocaCaracter: Essa classe foi desenvolvida para normalizar todos os caracteres que compõem o texto. Sendo assim, com o uso da função low da Biblioteca Strings, analisamos o texto e todos os caracteres que estão em maiúsculo são substituídos pelos seus correspondentes em minúsculo.
- Classe remPontuacao: Nesta classe com uso da função replace da Biblioteca Strings, realizamos a análise do texto e todos os caracteres especiais são excluídos.
- Classe remStopwords: Essa classe foi implementada a partir da função

stopwords da Biblioteca NLTK¹² e da lista de termos cadastradas como palavras não importantes do idioma Português também disponibilizada pela NLTK. Para isso, percorremos o texto a busca das palavras que estão contidas na lista, cada vez que essa palavra é encontrada no texto ela é excluída, restando no texto os demais termos que o compõem.

- Classe `remStopwEtiq`: Herdeira da classe `remStopwords`, essa classe é responsável por analisar os termos de acordo com sua classificação sintática. A classificação sintática é realizada pela classe `etiqSintatica` que será explicada mais à frente. A partir da análise das etiquetas de cada termo, verifica se a etiqueta está contida na lista de etiquetas não desejadas, se sim, descartamos a palavra. Os termos que permanecem no texto são aqueles classificados (etiquetados) como substantivos, verbos ou adjetivos.

6.1.2 Técnicas de pré-processamento do texto

Além das técnicas para limpeza dos textos, implementamos técnicas para realizar seu pré-processamento. Essas técnicas são explicadas a seguir.

6.1.2.1 Etiquetador Morfossintático

A técnica de Etiquetagem Morfossintática foi implementada através da classe `etiqSintatica`. Seu desenvolvimento deu-se a partir da utilização do Pacote NPLNet, versão 1.2.0. Este pacote contém classes e métodos responsáveis por executar a análise de uma frase e a partir da posição da palavra dentro da frase atribuir à palavra sua classificação sintática. Utilizamos do pacote a classe POSTagger para realizar a análise dos textos e atribuição das etiquetas sintáticas a cada termo do *corpus*.

12 A Biblioteca NLTK – Natural Language Toolkit, disponível em <http://nltk.org/>, é composta por algoritmos e técnicas para auxiliar no processamento de textos em aplicações escritas em Python.

6.1.2.2 Stemming

A técnica de Stemming tem por objetivo reduzir um termo a seu radical de formação, com isso o processo de busca por termos semelhantes é otimizado. A classe stemming foi criada a partir da classe RSLPStemmer da Biblioteca NLTK e do método stem.

Com o uso da classe disponibilizada pela NLTK, cada palavra do texto é analisada e comparada a um conjunto de regras. Cada regra corresponde a um tipo de variação possível para palavras escritas no idioma Português, representada por um afixo. A cada identificação de semelhança entre a palavra e uma regra, o afixo é retirado. Por fim, após a comparação com todas as regras, sobra o radical de formação da palavra que está em análise. Uma abstração do fluxo de atividades dessa técnica pode ser visto na Figura 14.

6.1.2.3 Normalização

A técnica de normalização visa a troca de termos sinônimos por uma única forma de representação dentro do texto. A classe normalizaTexto foi desenvolvida seguindo o fluxo de atividades apresentado na Figura 15.

A lista de sinônimos disponibilizada pelo Projeto TeP2.0 contém 19885 entradas que correspondem à relação de sinonímia de substantivos, verbos e adjetivos da Língua Portuguesa. A estrutura do arquivo é a seguinte:

```
263. [Verbo] {consentir, deixar, permitir} <973>
NUM1. [Tipo] {termos sinônimos} <NUM2>
[Tipo] = Classificação sintática
{termos sinônimos} = palavras que possuem sentidos semelhantes de
acordo com o contexto em que são utilizadas.
NUM1 = NUMERO DA LINHA DE REFERENCIA PARA TERMO SINÔNIMO
NUM2 = NUMERO DA LINHA DE REFERENCIA PARA TERMO ANTÔNIMO
```

Para agilizar o processo de busca, dividimos a lista de sinônimos em três listas menores, separadas de acordo com a classificação sintática das palavras, são elas: sinônimos_substantivos, sinônimos_verbos, sinônimos_adjetivos.

Para cada palavra do texto, primeiro verifica-se a sua classificação sintática. Se for uma palavra classificada como substantivo, a busca será realizada na lista de sinônimos_substantivos. Se for uma palavra classificada como verbo a busca dar-se

na lista `sinônimos_verbos`. E se for uma palavra classificada como adjetivo a busca é feita na lista `sinônimos_adjetivos`.

Caso a palavra seja encontrada na lista de sinônimos correspondente, a palavra será substituída pelo termo canônico que a representa. O termo canônico é definido ao encontrar pela primeira vez a lista de sinônimos referentes a um termo. Por exemplo, considerando que o substantivo “obra”, que está relacionado à lista de sinônimos “{feitura, obra, serviço}” também classificadas como substantivos, seja o primeiro termo desta lista de sinônimos a ser analisado. Logo, o “obra” será determinado como termo canônico representante desta lista de sinônimos. Caso os termos “feitura” e “serviço” sejam analisados, eles serão substituídos pelo termo canônico representante da lista de sinônimos a que pertencem, ou seja, serão substituídos pelo termo “obra”.

Por fim, temos o texto normalizado, com todas as relações de sinonímia identificadas e representadas pelo seu respectivo termo canônico.

6.1.3 Análise de Similaridade

Para verificar a semelhança entre as palavras que compõem um texto, implementamos na classe `calculaSimilaridade` a métrica indicada para calcular a similaridade entre palavras é denominada Similaridade de Cossenos.

A análise de similaridade foi desenvolvida a partir do método `cosine_similarity` que representa a métrica, pertencente ao pacote `metrics.pairwise` da Biblioteca `Scikit-Learn`¹³.

6.1.4 Modelos de Representação

Como modelos de representação computacional dos textos implementamos duas alternativas: as matrizes de relações semânticas criadas com a aplicação da técnica de Análise Semântica Latente (LSA) e o espaço vetorial criado a partir da utilização do Modelo Espaço Vetorial (SVM) e da medida TF-IDF.

13 Disponível em <http://scikit-learn.org/stable/>. Acesso em 07 de fevereiro de 2016.

6.1.4.1 Modelo LSA

A classe `modeloLSA` foi desenvolvida a partir da Biblioteca Gensim¹⁴, que possui todos os métodos necessários para aplicação da LSA em textos, grandes ou pequenos.

Sua implementação inicia-se com a limpeza do *corpus* que será utilizado como base para criação das matrizes de relação semântica dos termos. Esse *corpus* é composto pelos materiais didáticos enviados pelo mediador ou pelo texto de todas as argumentações que serão analisadas.

Sobre o *corpus* aplica-se uma métrica para calcular a frequência de repetição de um termo em todos os documentos que compõem o *corpus*. As palavras que aparecem uma única vez em todo o *corpus* são descartadas. Uma vez que essa técnica visa a criação de matrizes de relação a partir da coocorrência de termos em documentos diferentes, palavras que aparecem uma única não vez em todo o *corpus* não atendem a esse requisito.

O *corpus* passa a ser representado pelo modelo *bag-of-words*, no qual não é mais considerada a sequência em que as palavras aparecem num texto. Novamente, calcula-se a frequência dos termos nos documentos. Sobre as matrizes resultantes da análise de frequência aplica-se SVD.

Por fim, para tratar as matrizes resultantes da aplicação da SVD, utilizamos a classe MatrixSimilarity. Ao fim deste processo, temos como resultado as matrizes de relação que consideram a coocorrência dos termos num documento, o que indica que estes termos podem conter uma determinada relação semântica.

6.1.4.2 Modelo SVM

A classe `modeloSVM` foi criada a partir das classes CountVectorizer e TfidfTransformer pertencentes ao pacote `text` da Biblioteca Scikit-Learn.

Os textos são analisados e sobre eles é aplicado o cálculo para verificar a frequência dos termos (TF). A partir do valor obtido da frequência dos termos num documento, é calculado o valor inverso da frequência dos termos em relação a todos

14 Disponível em <http://radimrehurek.com/gensim/index.html>. Acesso em 07 de fevereiro de 2016.

os documentos do *corpus* (IDF). O valor TF é multiplicado pelo *log* do valor IDF. Isso descreve a aplicação direta da Fórmula 3.1, página 32.

Como resultado temos o Modelo Espaço-Vetorial composto pelo valor de importância de cada palavra que compõe o *corpus*.

6.1.5 Clusterização

A classe *criaGrupos* foi desenvolvida com base no algoritmo de clusterização K-Means. O algoritmo utilizado faz parte do módulo *Cluster*, da Biblioteca *SKLearn*. Para sua aplicação, como dito nos Capítulos 4 e 5, é necessário que os textos estejam representados por modelos computacionais.

A partir dos modelos, LSA ou SVM, o K-Means, selecionará K candidatos a centróides para iniciar os cálculos para definição dos grupos. A quantidade de grupos que serão criados é escolhido pelo professor ao acionar o recurso Grupos de Similaridade. Logo, o valor de K é definido pelo usuário do sistema. Caso a classe *criaGrupos* seja utilizada por outro recurso, o número de grupos e centróides fica definido como $K = 3$.

Como diferencial, essa implementação do K-Means permite que seja realizada, antes da escolha dos centróides, o cálculo prévio da distância entre os documentos que serão agrupados, para orientar a escolha dos centróides. Com os centróides definidos, é aplicado o cálculo da Distância Euclidiana entre o centróide e demais documentos do *corpus*. O cálculo é repetido até que a condição de parada seja atingida. No nosso caso, a partir de experimentos, a condição de parada foi definida como cem mil iterações.

Por fim, o *corpus* é subdividido em grupos de semelhança, onde os textos pertencentes a um mesmo grupo possuem um nível de semelhança maior entre si do que com os demais textos que compõem o *corpus*.

6.2 RECURSOS DO PROTÓTIPO

Os recursos disponíveis nessa versão do protótipo fazem uso de todas as funcionalidades do Núcleo de Recuperação de Informações. A seguir detalhamos os recursos contidos no protótipo.

6.2.1 Recurso Grupos de Similaridade

O recurso Grupos de Similaridade foi implementado conforme sua descrição na subseção 5.1.2.1. Logo, para seu funcionamento é necessário que o mediador informe quantos grupos deseja que sejam criados e qual argumentação será utilizada como base para a criação dos grupos, ou seja, o mediador deve escolher entre a argumentação inicial e argumentação final. Após realizadas as escolhas, o mediador deve clicar no botão “Criar grupos” para iniciar o processo do recurso.

Em seguida, o sistema resgatará os textos da argumentação escolhida do banco de dados do Sistema “Debate de Teses” (SDT), processá-los com o uso das técnicas de limpeza de *corpus* e técnicas de pré-processamento, representar os textos com os modelos de representação e, por fim, utilizar a clusterização para criar os grupos. Finalmente, os grupos são apresentados ao mediador.

6.2.2 Recurso Síntese das Autorias

Este recurso foi desenvolvido de acordo com sua descrição na Seção 5.1.2.6, para apresentar ao mediador os termos mais importantes escritos pelos alunos na fase de Argumentação Final.

Os textos das argumentações finais são recuperados do banco de dados do SDT, sobre os quais aplicam-se técnicas para limpeza dos textos, técnicas de pré-processamento, SVM para representar como modelo de representação dos textos e a clusterização para criação de grupos de semelhança.

Após a criação dos grupos, aplicamos a medida TF-IDF nas argumentações finais dos alunos que fazem parte de um mesmo grupo para identificar quais os termos mais relevantes. Os termos identificados são apresentados na forma de nuvem para

o mediador.

6.3 CONSIDERAÇÕES FINAIS DO CAPÍTULO

Neste capítulo apresentamos o processo de construção do protótipo computacional do Sistema Alpes, com objetivo de validar o Núcleo de Recuperação de Informações (NRI) proposto na Seção 5.2.

O desenvolvimento das técnicas e algoritmos que compõem o NRI demonstraram-se suficientes para alcançar os objetivos propostos e esperados de cada recurso e agente contido no Alpes.

A fim de comprovar que a utilização do Alpes mostra-se como uma solução favorável para que mediações pedagógicas sejam aplicadas em paralelo às arquiteturas pedagógicas desenvolvidas através de ambientes digitais, aplicamos o recurso Grupos de Argumentação a um caso real, conforme relatamos no Capítulo 7.

7 APLICAÇÃO DA PROVA DE CONCEITO A UM “DEBATE DE TESES”

Com o objetivo de comprovarmos a utilização da prova de conceito desenvolvida como facilitador do processo de mediação pedagógica, comparamos o processo de criação manual de grupos com a aplicação do Recurso Grupos de Similaridade, ambos a partir da análise da argumentação inicial produzida pelos alunos.

Os textos analisados foram produzidos com a aplicação da Arquitetura Pedagógica “Debate de Teses” (APDT) como atividade pré-curso do curso de extensão XIV Maratona de Empreendedorismo da UFRGS¹⁵ realizado em 2013 (MICHELS, 2014).

Os objetivos de utilização da APDT destacados por Michels (2014) foram:

[...] (1) verificar os conhecimentos prévios dos alunos em relação aos temas abordados nas teses; (2) possibilitar a construção de conhecimento acerca das teses; (3) propiciar uma exploração individual e coletiva sobre as teses; (4) oportunizar um espaço de reflexão e experimentação de estratégias de *feedback* e contra-argumentação, facilitando a elaboração do modelo de negócios e do plano de negócios (MICHELS, 2014. p. 67).

O SDT foi utilizado como suporte computacional para promover a aplicação da APDT e assim possibilitar a construção de conhecimento cooperativa entre os participantes, sobre os temas Modelo de Negócios e Plano de Negócios. Para isso, foram propostas três teses para o debate, são elas (MICHELS, 2014):

- [1] “O mundo mudou, o perfil do consumidor mudou, e, conseqüentemente, a forma de se planejar negócios também mudou”
- [2] “O plano de negócios aumenta a probabilidade de sucesso dos negócios”
- [3] “A visão macro do negócio é desenvolvida com o auxílio do modelo de negócios e a visão micro com o auxílio do plano de negócios. Logo, o empreendedor precisa ter visão do todo e, se necessário, a visão de cada elemento do negócio”

Do total de participantes do debate promovido por Michels (2014), utilizamos para fins de aplicação e coleta de dados, uma amostra composta pelos 21 alunos que concluíram todas as etapas propostas pela APDT. Uma vez que, além da aplicação do protótipo os grupos também foram criados manualmente, optamos por utilizar uma amostra menor devido ao tempo de análise e realização da tarefa. Sendo assim, os agrupamentos foram realizadas sobre as argumentações iniciais produzidas sobre a Tese [1] “O mundo mudou, o perfil do consumidor mudou, e,

15 Debate completo disponível em <http://www.pead.faced.ufrgs.br/sites/cms/debate/index.php?&idmask=&idgrupo=1064&idpagina=14&paginaAnterior=&>. Acesso em 25 de fevereiro de 2016.

consequentemente, a forma de se planejar negócios também mudou”.

Para garantir o anonimato dos alunos que compõem a amostra, organizamos os 21 alunos em ordem alfabética e iremos nos referir a eles como ‘A01’, ‘A02’, ‘A03’,..., ‘A19’, ‘A20’, ‘A21’.

Deste modo, detalhamos na Seção 7.1 as duas fases do experimento realizado. Na Seção 7.1.3 apresentamos as análises dos resultados obtidos com o experimento. E, na Seção 7.2 trazemos nossas considerações finais sobre o capítulo.

7.1 EXPERIMENTO DE APLICAÇÃO

Nesta seção apresentamos os detalhes do experimento realizado para fins de validação do recurso Grupos de Similaridade. Para isso, o experimento foi dividido três fases. Na primeira realizamos a divisão manual dos grupos. Na segunda, utilizamos o recursos para criação dos grupos. Na última fase do experimento comparamos os grupos criados na fase 1 com os grupos criados na fase 2, expondo nossa análise sobre os resultados obtidos.

7.1.1 Fase 1: Divisão manual dos grupos

Para criar os grupos, foi necessário simular o trabalho que o mediador teria para fazê-lo. Logo, acessamos a página correspondente ao perfil do aluno dentro do Sistema “Debate de Teses”, para cada um dos 21 alunos da amostra. Na página do aluno, temos acesso à argumentação inicial do aluno. A primeira fase do experimento, consistiu em dividir os 21 alunos em 3 grupos. Para facilitar a divisão dos grupos, criamos uma tabela¹⁶ e nela colocados todas as argumentações escritas pelos 21 alunos.

A partir disso, iniciamos a leitura pela argumentação inicial produzida pelo aluno A01, utilizando esta argumentação como base para criar o Grupo1. Finalizada a leitura da argumentação do aluno A01, iniciamos a leitura da argumentação do aluno A02, percebemos que as duas argumentações não eram semelhantes, colocando

¹⁶ A tabela de apoio pode ser acessa em <https://docs.google.com/spreadsheets/d/1giSbqyZirJUix-rpOD8marzipatqCoWDIH82Rv3w0Yw/edit?usp=sharing>. Acesso em 26 de fevereiro de 2016.

A01 e A02 em grupos separados. Com a leitura da argumentação do A03, vimos que o texto possui semelhança com o texto do A01, colocando os dois juntos. A análise e agrupamentos dos alunos repetiu-se até todos os alunos terem sido colocados em um dos grupos.

No Quadro 2 apresentamos a distribuição realizada nesta primeira etapa do experimento, onde a primeira coluna representa o Grupo1 e seus alunos, a segunda coluna mostra o Grupo2 e seus alunos e a terceira coluna mostra o Grupo3 e seus respectivos alunos.

Quadro 2 - Divisão da amostra em três grupos

Grupo1	Grupo2	Grupo3
A01	A02	A07
A03	A04	A16
A05	A08	A21
A06	A10	
A09	A11	
A12	A13	
A17	A14	
A18	A15	
A19	A20	

Fonte: Autoria Própria

Para terminar a primeira fase do experimento e criar os três grupos gastamos cerca de 1h30. Nesta primeira etapa, destacamos que a criação dos grupos é cansativa e trabalhosa. Ao iniciar a análise da argumentação do A15, os pontos principais das argumentações lidas anteriormente tinham sido esquecidos, o que nos levou a reler parcialmente as argumentações que já estavam agrupadas para dar continuidade na atividade. Além disso, para o tamanho da amostra, percebemos que a criação de apenas três grupos não é suficiente, uma vez que alguns agrupamentos foram feitos mesmo tendo uma baixa semelhança entre as argumentações.

A segunda fase do experimento consistiu na divisão dos 21 alunos em quatro grupos de semelhança. Para isso, repetimos o mesmo processo realizado na primeira fase do experimento, iniciando a análise pela argumentação produzida pelo A01. O processo se repetiu até todas as argumentações terem sido analisadas e os alunos

agrupados. Esta fase demorou cerca de 1 h para ser concluída. Notamos que uma vez que já havíamos lido as argumentações, a leitura foi mais rápida e tranquila, facilitando o processo de análise e agrupamento.

No Quadro 3 apresentamos a divisão feita nesta segunda fase, onde temos na primeira coluna o Grupo1 com seus respectivos alunos, na segunda coluna o Grupo2 com seus alunos, na terceira coluna o Grupo3 e seus alunos, e por fim, na quarta coluna o Grupo4 com seus alunos.

Quadro 3 - Divisão da amostra em quatro grupos

Grupo1	Grupo2	Grupo3	Grupo4
A01	A02	A06	A07
A03	A04	A19	A16
A05	A08		A21
A09	A10		
A12	A11		
A17	A13		
A18	A14		
	A15		
	A20		

Fonte: Autoria Própria

Na fase final do experimento, dividimos a amostra em 5 grupos de semelhança. Como na segunda fase o tempo para criação dos grupos diminuiu, sendo gastos 40 min para sua conclusão.

O processo de análise foi realizado ao contrário dessa vez, ou seja, a análise foi iniciada a partir do A21. Essa troca foi necessária para evitar que, inconscientemente, criássemos grupos próximos aos que já tínhamos criado nas fases anteriores por seguirmos a mesma ordem de leitura e distribuição dos grupos.

No Quadro 4 temos a divisão desta última fase do experimento. Nas colunas estão representados os grupos e os alunos que pertencem a cada grupo.

Quadro 4 - Divisão da amostra em cinco grupos

Grupo1	Grupo2	Grupo3	Grupo4	Grupo5
A21	A20	A19	A15	A18
A16	A14	A03	A04	A17
A07	A13			A12
A01	A10			A11
	A08			A09
	A02			A06
				A05

Fonte: Autoria Própria

Por fim, concluímos que o experimento foi muito cansativo. Ler e reler as argumentações produzidas pelos alunos despence uma quantidade de tempo considerável, ao todo foram mais 3 h para conclusão de todas as fases do experimento. Além disso, nos sentimos condicionado a criar grupos com uma quantidade próxima de alunos para facilitar as intervenções. E constatamos que seria inviável fazer o mesmo processo para as demais teses, dado o tempo gasto e o cansaço que a atividade proporciona.

7.1.2 Fase 2: Uso do Recurso Grupos de Similaridade

Este recurso foi desenvolvido para analisar os textos produzidos nas etapas de argumentação, e a partir dessas análises dividir os alunos em grupos de semelhança. Conforme relatamos na Seção 6.2.1 é necessário que seja informado o número de grupos e qual argumentação será analisada. Para comparar os resultados produzidos com a criação manual dos grupos, o recurso foi acionado três vezes para divisão da amostra em três, quatro e cinco grupos a partir da análise da argumentação inicial.

Os grupos formados pelo recurso são apresentados de forma gráfica para o mediador. Contudo, para facilitar a relação entre os grupos formados pelo especialista e os grupos formados pelo recursos, organizamos os grupos formados com a aplicação do recurso seguindo a mesma regra de formação utilizada na divisão manual.

Logo, na primeira e segunda divisões realizadas pelo recurso, damos o nome de Grupo1 para grupo que contém o A01, e de Grupo2 para o grupo que contém o A02, e assim por diante. Na última distribuição, damos o nome de Grupo1 para o grupo que contém o A21, e de Grupo2 para o grupo que contém o aluno A20, e assim por diante.

O resultado obtido com a aplicação do recurso para divisão da amostra em três grupos de semelhança é apresentado no Quadro 5. Como podemos observar, na primeira coluna temos o Grupo1 e todos os alunos que o compõem, na segunda coluna o Grupo2 com seus respectivos alunos, e na coluna três o Grupo3 com seus respectivos alunos.

Quadro 5 - Divisão da amostra em três grupos com o uso do Recurso Grupos de Similaridade

Grupo1	Grupo2	Grupo3
A01	A02	A04
A03	A07	A06
A05	A16	A08
A17	A20	A09
A18	A21	A10
A19		A11
		A12
		A13
		A14
		A15

Fonte: Autoria Própria

Em sequência, acionamos novamente o recurso, selecionando a Argumentação Inicial como etapa a ser analisada, e informando que devem ser criados quatro grupos. O resultado apresentado pelo recurso está no Quadro 6, onde cada coluna do quadro representa um grupo.

Quadro 6 - Divisão da amostra em quatro grupos com o uso do Recurso Grupos de Similaridade

Grupo1	Grupo2	Grupo3	Grupo4
A01	A02	A04	A06
A03	A07	A08	A11
A05	A16	A10	A12
A09	A21	A13	
A17		A14	
A18		A15	
A19		A20	

Fonte: Autoria Própria

Para finalizar o experimento, utilizamos o recurso para a divisão da amostra em cinco grupos de semelhança a partir da análise da Argumentação Inicial. No Quadro 7 apresentamos o resultado deste último teste, onde cada coluna do quadro representa um grupo com seus respectivos alunos.

Quadro 7 - Divisão da amostra em cinco grupos com o uso do Recurso Grupos de Similaridade

Grupo1	Grupo2	Grupo3	Grupo4	Grupo5
A21	A20	A19	A18	A12
A16	A15	A03	A17	A11
A07	A14	A01	A09	A06
A02	A13		A05	
	A10			
	A08			
	A04			

Fonte: Autoria Própria

Ao todo, não foram gastos mais do que 10 min para que o Recurso Grupos de Similaridade apresentasse as divisões da amostra em três, quatro e cinco grupos.

7.1.3 Fase 3: Análise dos Resultados

O objetivo principal do recurso Grupo de Similaridade é facilitar a prática da mediação pedagógica “Agrupar indivíduos que possuem argumentações semelhantes” (página 26). A formação dos grupos a partir da análise da argumentação inicial auxilia o mediador a identificar quais alunos possuem opiniões

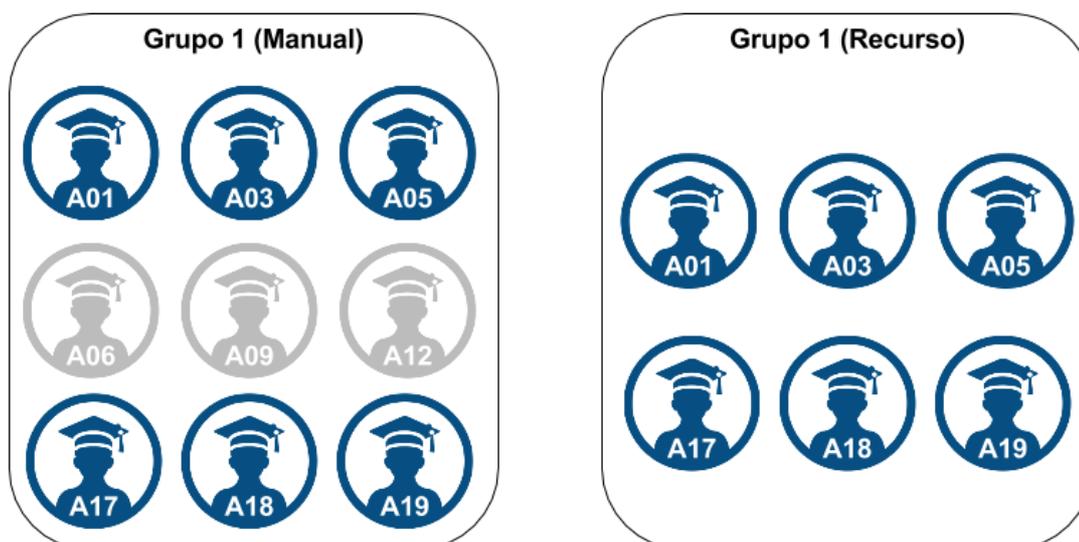
semelhantes sobre a tese no início do debate. Outrossim ajuda a análise do nível de conhecimento dos alunos nesta fase inicial.

Ao compararmos os resultados obtidos com a criação dos grupos de forma manual com a criação dos grupos com a utilização do recurso, chegamos às seguintes conclusões:

A. Para divisão da amostra em três grupos:

- I. O Grupo1 criado manualmente tem 9 alunos e o Grupo1 formado pelo recurso tem 6 alunos. Sendo que todos os alunos do Grupo1 criado pelo recurso fazem parte do Grupo1 criado manualmente. Sendo assim, a semelhança entre as duas distribuições chega a 66,7%. A Figura 21 apresenta os dois grupos com destaque para os alunos que fazem parte de ambos os grupos.

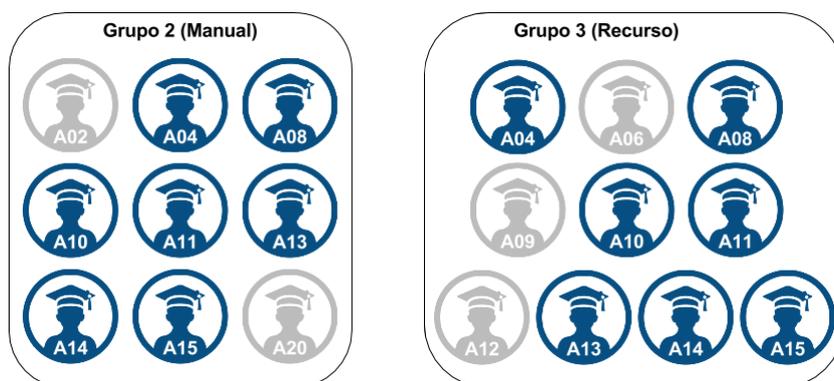
Figura 21 - Divisão da amostra em três grupos - Grupo1 (Manual) e Grupo1 (Recurso)



Fonte: Autoria Própria

- II. O Grupo2 criado manualmente tem 9 alunos possui maior semelhança com o Grupo3 criado pelo recurso que contém 10 alunos. Os dois grupos contêm o subgrupo de alunos destacados na Figura 22 como membros, o que nos mostra uma relação de 70% de semelhança entre os grupos.

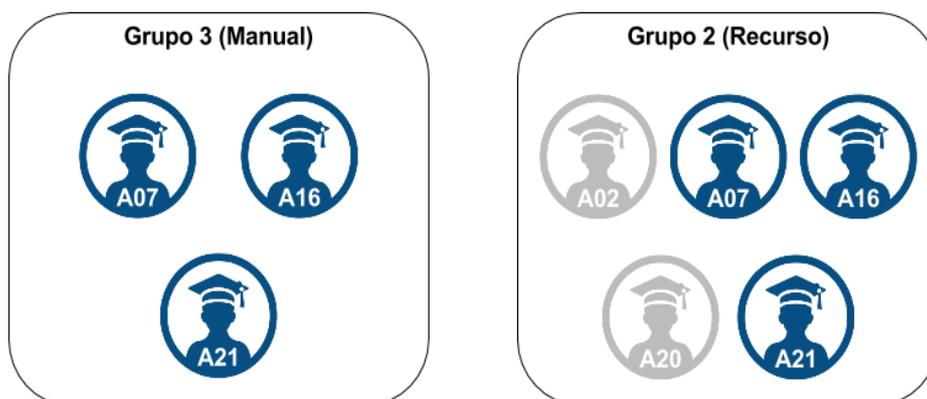
Figura 22 - Divisão da amostra em três grupos - Grupo2 (Manual) e Grupo3 (Recurso)



Fonte: Autoria Própria

III.O Grupo3 criado manualmente possui uma relação de 60% de semelhança com Grupo2 criado pelo recurso. A Figura 23 destaca a relação entre os dois grupos.

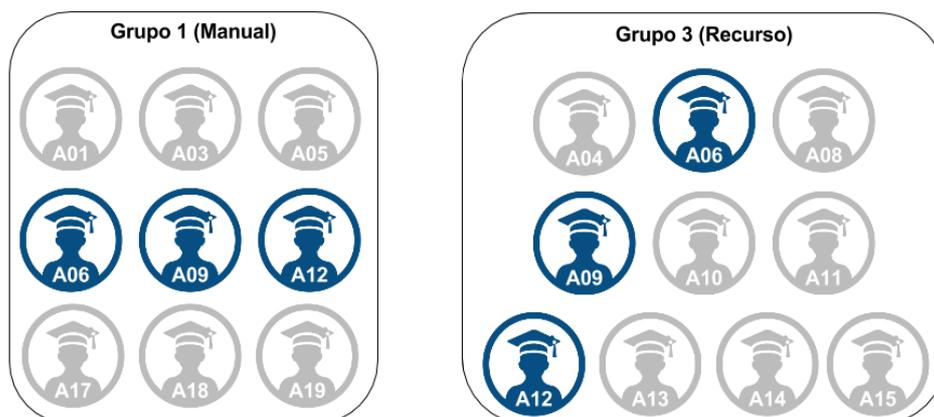
Figura 23 - Divisão da amostra em três grupos - Grupo 3 (Manual) e Grupo 2 (Recurso)



Fonte: Autoria Própria

IV.O Grupo1 criado manualmente possui uma relação de 30% de semelhança com o Grupo3 criado pelo recurso, pois nos dois estão contidos os alunos A06, A09 e A12 (Figura 24).

Figura 24 - Divisão da amostra em três grupos - Grupo1 (Manual) e Grupo3 (Recurso)

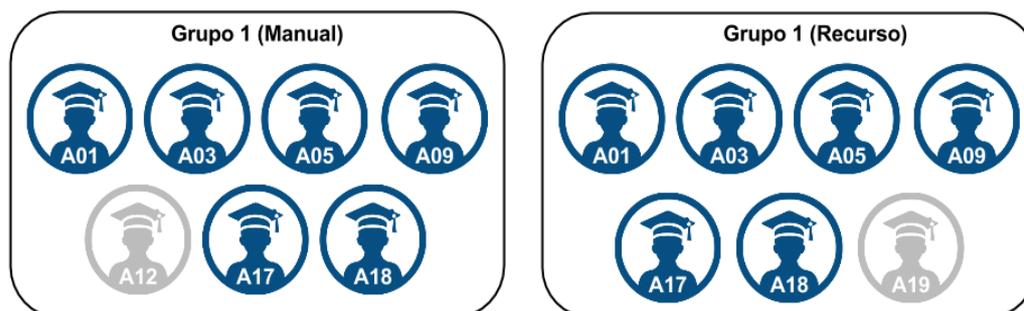


Fonte: Autoria Própria

B. Para divisão da amostra em quatro grupos:

- I. O Grupo1 criado manualmente possui uma semelhança de 85,7% com o Grupo1 criado pelo recurso (Figura 25).

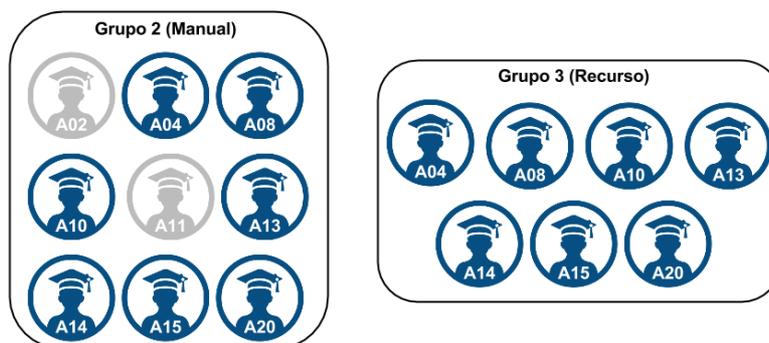
Figura 25 - Divisão da amostra em quatro grupos - Grupo1 (Manual) e Grupo1 (Recurso)



Fonte: Autoria Própria

- II. O Grupo2 criado manualmente possui uma relação de 77,8% com o Grupo3 criado pelo recurso. No agrupamento manual os alunos A02 e A11 foram alocados com os demais alunos no Grupo2, sendo essa diferença entre os membros dos grupos (Figura 26).

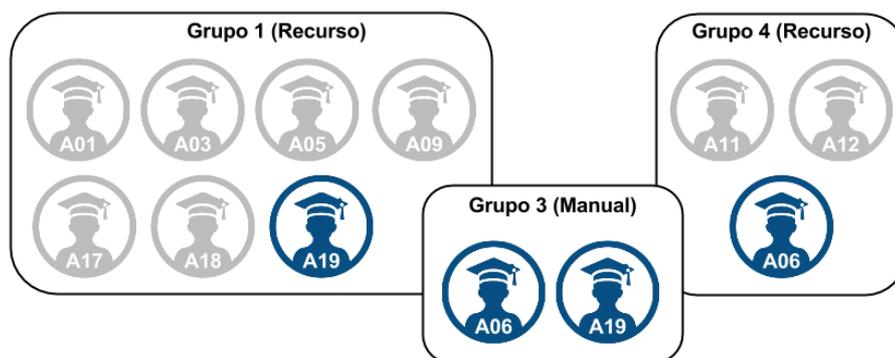
Figura 26 - Divisão da amostra em quatro grupos - Grupo2 (Manual) e Grupo3 (Recurso)



Fonte: Autoria Própria

III. O Grupo3 criado manualmente possui apenas dois alunos (A06 e A19), como podemos observar na Figura 27. Os dois alunos foram distribuídos em grupos distintos pelo recurso, que nos mostra que a relação de semelhança entre o Grupo3 (Manual) e o Grupo1 (Recurso) é de apenas 14,3%. Já a semelhança entre o Grupo3 (Manual) e o Grupo4 (Recurso) é de 33,3%, isso porque ambos os grupos possuem poucos alunos

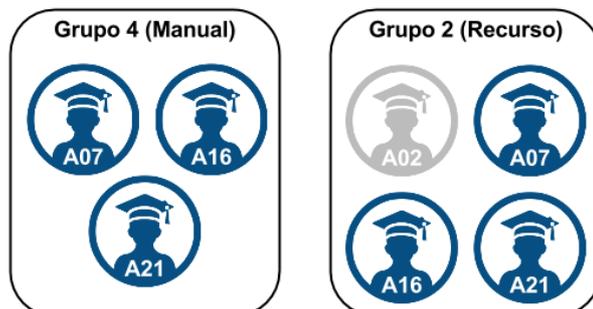
Figura 27 - Divisão da amostra em quatro grupos - Relação entre Grupo3 (Manual), Grupo1 (Recurso) e Grupo4 (Recurso)



Fonte: Autoria Própria

IV. O Grupo4 criado manualmente possui uma semelhança de 75% com o Grupo2 criado pelo recurso. Na Figura 28 os dois grupos são apresentados.

Figura 28 - Divisão da amostra em quatro grupos - Grupo4 (Manual) e Grupo2 (Recurso)



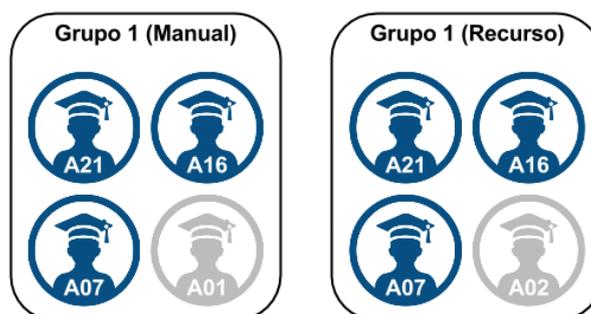
Fonte: Autoria Própria

V. Cabe ressaltar, que as demais semelhanças entre os grupos são irrelevantes. E, que nesta distribuição da amostra, o caso mais discrepante ao compararmos os grupos separados manualmente com os grupos resultantes da aplicação do recurso, está na distribuição do Grupo3 (Manual), que não possui uma boa relação com nenhum grupo criado pelo recurso.

C. Para divisão da amostra em cinco grupos:

I. Na Figura 29 temos o Grupo1 (Manual) e o Grupo1 (Recurso), na qual destacamos em azul os alunos que fazem parte de ambos os grupos, o que representa uma relação de semelhança de 75% entre os dois grupos.

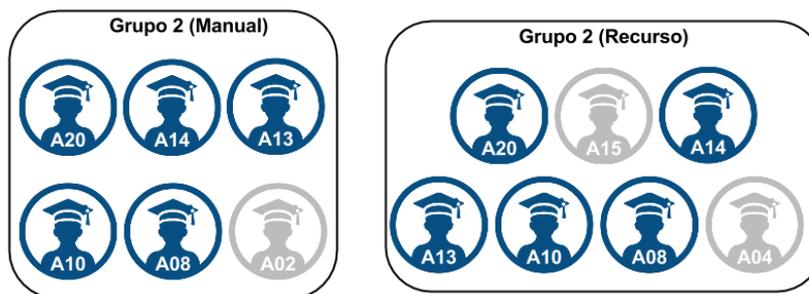
Figura 29 - Divisão da amostra em cinco grupos - Grupo1 (Manual) e Grupo1 (Recurso)



Fonte: Autoria Própria

II. Como podemos observar na Figura 30, o Grupo2 (Manual) possui uma relação de semelhança entre seus membros de 71,4% com o Grupo2 (Recurso).

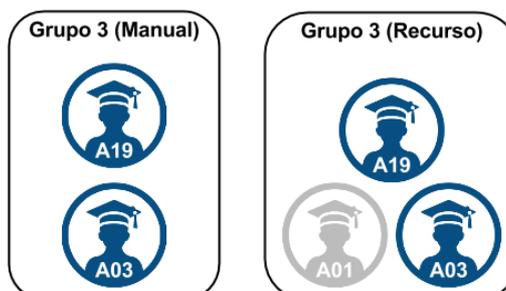
Figura 30 - Divisão da amostra em cinco grupos - Grupo2 (Manual) e Grupo2 (Recurso)



Fonte: Autoria Própria

III. Apesar de ter apenas dois membros, o Grupo3 (Manual) possui relação de 66,7% com o Grupo3 (Recurso), como podemos observar na Figura 31.

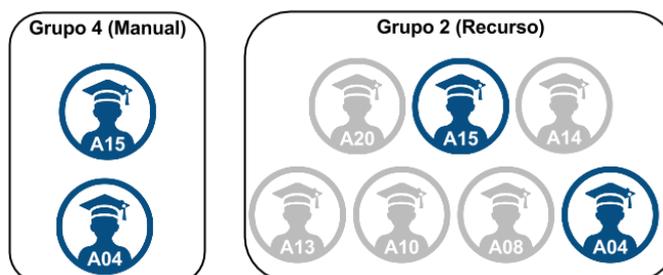
Figura 31 - Divisão da amostra em cinco grupos - Grupo3 (Manual) e Grupo3 (Recurso)



Fonte: Autoria Própria

IV. A relação de semelhança do Grupo4 (Manual) é de 28,6% com o Grupo2 (Recurso), pois seus dois membros fazem parte do Grupo2 (Recurso). Podemos observar essa relação através da Figura 32.

Figura 32 - Divisão da amostra em cinco grupos - Grupo4 (Manual) e Grupo2 (Recurso)

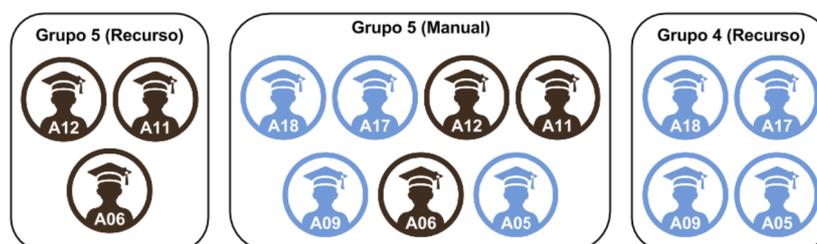


Fonte: Autoria Própria

V. Analisando a Figura 33 percebemos que o Grupo5 (Manual) foi dividido em dois grupos pelo recurso, Grupo4 (Recurso) e Grupo5 (Recurso). A

relação de semelhança do Grupo5 (Manual) com o Grupo4 (Recurso) é de 57,1%, e do Grupo5 (Manual) com o Grupo5 (Recurso) é de 42,9%.

Figura 33 - Divisão da amostra em cinco grupos - Relação entre Grupo5 (Manual), Grupo4 (Recurso) e Grupo5 (Recurso)



Fonte: Autoria Própria

VI. Por fim, destacamos que essa última fase do experimento mostrou que a distribuição dos alunos A01 e A02 foi distinta ao compararmos o agrupamento manual e o agrupamento realizado pelo recurso.

Ao finalizar as comparações entre os dados coletados com os experimentos, podemos observar (Tabela 1) que os agrupamentos realizados com a utilização do recurso obtiveram um nível satisfatório de semelhança nas distribuições dos alunos comparadas às distribuições manuais.

Tabela 1 - Semelhanças entre as formações dos grupos

DIVISÃO	MANUAL X RECURSO	% SEMELHANÇA
3 GRUPOS	GRUPO1 x GRUPO1	66,7%
3 GRUPOS	GRUPO2 x GRUPO3	70,0%
3 GRUPOS	GRUPO3 x GRUPO2	60,0%
4 GRUPOS	GRUPO1 x GRUPO1	85,7%
4 GRUPOS	GRUPO2 x GRUPO3	77,8%
4 GRUPOS	GRUPO4 x GRUPO2	75,0%
5 GRUPOS	GRUPO1 x GRUPO1	75,0%
5 GRUPOS	GRUPO2 x GRUPO2	71,4%
5 GRUPOS	GRUPO3 x GRUPO3	66,7%
5 GRUPOS	GRUPO5 x GRUPO5	42,9%
5 GRUPOS	GRUPO5 x GRUPO4	57,1%

Fonte: Autoria Própria

Na Tabela 1 temos como destaque as distribuições que alcançaram mais de 60% de semelhança com a distribuição manual e também as distribuições que correspondem a subgrupos dos grupos formados manualmente.

7.2 CONSIDERAÇÕES FINAIS DO CAPÍTULO

A realização do experimento para validar a implementação feita do Recurso Grupos de Similaridade foi necessária para comprovar a eficácia de aplicação do recurso. Algumas comparações são inevitáveis, como o tempo gasto na Fase 1 e Fase 2 do experimento. Na Fase 1 demoramos cerca de 3 h para criar os grupos, na Fase 2 com a aplicação do recursos os grupos foram definidos e apresentados em menos de 10 min, isso considerando as três etapas de cada fase.

Ademais, compreendemos que os grupos criados na Fase 1 possuem critérios de distribuição distintos dos grupos criados na Fase 2 com a aplicação do recurso, isso porque ao analisarmos os textos utilizamos nossos conhecimentos prévios sobre o contexto em que os textos foram escritos, fazendo uma análise semântica e uma comparação entre o conteúdo dos textos analisados. Já com a aplicação do recurso a análise realizada leva em consideração a semelhança sintática entre os textos, baseado no valor de semelhança encontrado com a aplicação das métricas e algoritmos que compõem o recurso (cf. Seção 6.2.1).

Apesar de seguirem critérios distintos durante seus processos de criação obtivemos um total de 75% de semelhança entre os grupos formados na Fase 1 comparados aos grupos formados na Fase 2.

Desta maneira entendemos que o experimento atesta a eficácia de uso do recurso e evidencia uma nova possibilidade para tratar as informações produzidas pelos alunos em suas interações, com o intuito de facilitar a aplicabilidade das Mediações Pedagógicas em Ambientes Virtuais de Ensino e Aprendizagem.

8 CONCLUSÕES

Iniciamos as conclusões voltando às hipóteses de pesquisa descritas no Capítulo 1.

Com o experimento realizado e descrito no Capítulo 7 concluímos que ao utilizar recursos computacionais para agilizar a execução de algumas atividades, o professor poderá utilizar seu tempo e esforço em outras tarefas que necessitem de sua atenção. Dessa forma, confirmamos a hipótese que o uso de suporte computacional é uma boa alternativa para apoiar as atividades do professor em ambientes virtuais de aprendizagem.

Ainda com apoio do experimento, conseguimos confirmar parcialmente, a hipótese de que o uso desse suporte computacional viabiliza que o professor desenvolva mediações pedagógicas. Em parte, pois a prova de conceito implementada não foi utilizada junto de uma aplicação em tempo real da Arquitetura Pedagógica “Debate de Teses”. Contudo, acreditamos que com os resultados obtidos com a aplicação do Recurso Grupos de Similaridade para a análise de um debate já realizado o professor tem suporte para aplicação de, pelo menos, três mediações pedagógicas descritas na Seção 2.3.

Em relação à hipótese que versa sobre o processamento de textos baseado na sintaxe ser suficiente para destacar semelhanças entre textos e com isso fornecer informações que facilitem o emprego de mediações pedagógicas em ambientes virtuais, verificamos que essa hipótese se confirma a partir da concepção do Núcleo de Recuperação de Informações (Seção 5.2). Parte central da versão atual do protótipo, o núcleo desenvolvido foi capaz de correlacionar os textos produzidos pelos alunos e agrupá-los a partir da análise de semelhança entre eles. Sendo assim, as técnicas em nível de análise sintática que compõem o núcleo foram eficientes na recuperação de informações importantes dos textos, na relação de semelhança destas informações e na apresentação dos grupos de semelhança requeridos pelo recurso Grupos de Similaridade.

Sobre o objetivo geral norteador desta pesquisa (Seção 1.3), ao revisitarmos a concepção da proposta de solução, o desenvolvimento do protótipo e a condução do experimento, bem como a análise dos resultados obtidos, concluímos que

conseguimos demonstrar que o uso de suporte computacional pode ajudar o professor na execução de suas atividades, reduzindo seu esforço e tempo despendido com as mesmas. Além disso, o suporte computacional resultante desta pesquisa, proporciona a possibilidade de desenvolver novas estratégias pedagógicas e aplicá-las em conjunto às arquiteturas pedagógicas diversas, bem como pode auxiliar na prática das mediações pedagógicas conduzidas em ambientes virtuais de ensino e aprendizagem.

8.1 PRODUÇÃO CIENTÍFICA

No decorrer desta pesquisa, elaboramos três artigos científicos que foram submetidos, aprovados e apresentados em dois eventos de referência que congregam as produções científicas, a nível nacional e internacional, sobre as pesquisas em Informática na Educação.

O primeiro artigo intitulado *“ALPES: Um sistema multiagentes para análise de produções textuais no contexto de um Debate de Teses”* foi apresentado e publicado no XXV Simpósio Brasileiro de Informática na Educação no ano de 2014. Este artigo contém nossas concepções iniciais sobre esta pesquisa. Nele detalhamos as mediações pedagógicas que havíamos identificado na época, bem como nossa ideia inicial sobre como a solução seria desenvolvida.

O segundo artigo, com o título *“O suporte computacional como auxílio a mediação pedagógica em um Debate de Teses”* foi apresentado e publicado na XIX Conferência Internacional sobre Informática na Educação, também no ano 2014. Neste artigo o foco foi o processo de mediação pedagógica, onde destacamos sua importância para auxiliar na construção de conhecimento individual e coletivo dos alunos. Além de destacamos de forma geral a concepção inicial da solução para auxiliar na prática das mediações pedagógicas.

Com a pesquisa mais avançada, elaboramos o artigo *“Apoio a mediação pedagógica em um Debate de Teses utilizando técnicas de processamento de texto”* que foi apresentado e publicado no XXVI Simpósio Brasileiro de Informática na Educação no ano de 2015. Neste último artigo, nosso foco foi na descrição do núcleo

computacional da solução, onde descrevemos o desenvolvimento de parte do núcleo de recuperação de informações, bem como, sua aplicação a um conjunto de dados, a coleta e análises dos dados, e os resultados alcançados com sua aplicação.

8.2 TRABALHOS FUTUROS

Como trabalhos futuros, destacamos como proposta de melhoria para a versão descrita nesta pesquisa, a incorporação de técnicas de Processamento de Linguagem Natural que deem suporte à análise semântica dos textos, com aporte em ontologias de domínio que podem ser predefinidas no sistema, ou que poderão ser construídas pelo professor através de um módulo específico.

Outra melhoria seria acrescentar uma base de conhecimento monitorada e manipulada por um agente cognitivo que tem como papel analisar as informações contidas em debates distintos que pertençam a um mesmo domínio do conhecimento, e armazenar nesta base afirmações e/ou conceitos apresentados pelas teses, argumentações e demais textos produzidos nas interações do debate.

Além disso, a remodelagem do agente de contexto como um agente cognitivo, que tenha como base de conhecimento as ontologias de domínio, para monitorar em tempo real a produção dos textos feitos pelos alunos, fazendo indicações de melhorias e ou apontamentos para garantir que as produções ocorrem dentro do tema proposto pela tese.

Outro trabalho futuro é o desenvolvimento completo do Alpes e sua disponibilização para uso junto aos dois sistemas que dão suporte computacional para o desenvolvimento da Arquitetura Pedagógica “Debate de Teses”, possibilitando sua aplicação em tempo real como suporte a um debate que esteja em desenvolvimento.

Por fim, seria interessante criar uma abstração do núcleo de processamento de informações para apoiar a mediação em conjunto com as demais arquiteturas pedagógicas que utilizam a produção de textos e conversações como base para suas interações.

REFERÊNCIAS

ALUÍSIO, Sandra et al. An account of the challenge of tagging a reference corpus for brazilian portuguese. In: **Computational Processing of the Portuguese Language**. Springer Berlin Heidelberg, 2003. p. 110-117.

ÁVILA, Ricardo L.F. de; SOARES, José M. Uso de técnicas de pré-processamento textual e algoritmos de comparação como suporte à correção de questões dissertativas: experimentos, análises e contribuições. In: **Anais do Simpósio Brasileiro de Informática na Educação**. 2013. p. 727.

AZEVEDO, Breno F. T.; BEHAR, Patricia A.; REATEGUI, Eliseo B. Análise das mensagens de fóruns de discussão através de um software para mineração de textos. In: **Anais do Simpósio Brasileiro de Informática na Educação**. 2011.

CARVALHO, Marie Jane S.; NEVADO, Rosane A.; MENEZES, Crediné S. de. Arquiteturas pedagógicas para educação a distância. **Aprendizagem em rede na educação a distância: estudos e recursos para formação de professores**. Porto Alegre: Ricardo Lenz, v. 1, p. 36-52, 2007.

CARVALHO, Felipe G. de. **Comportamento em Grupo de Personagens do Tipo Black&White**. 2004. Dissertação de Mestrado, Pontifícia Universidade Católica do Rio de Janeiro, Rio de Janeiro.

CASTRO, Jaelson; ALENCAR, Fernanda; SILVA, Carla. Engenharia de software orientada a agentes. **Atualizações em Informática**, p. 245-282, 2006.

COPPIN, Ben. **Artificial intelligence illuminated**. Jones & Bartlett Learning, 2004.

DIAS-DA-SILVA, Bento C. et al. A construção de um thesaurus eletrônico para o português do Brasil. In: **International Joint Conference 7th Iberoamerican Conference on Artificial Intelligence, 15th Brazilian Conference on Artificial Intelligence and V PROPOR**, 2000. p. 1-11.

DJANGO 1.6. 2014. Disponível em: <<https://www.djangoproject.com/>>. Acesso em 10 jan. 2016.

DUMAIS, Susan T. Latent semantic analysis. **Annual review of information science and technology**, v. 38, n. 1, p. 188-230, 2004.

FELDMAN, Ronen; SANGER, James. **The text mining handbook: advanced approaches in analyzing unstructured data**. Cambridge University Press, 2007.

FERNANDES JUNIOR, José I. C.; MENEZES, Crediné S. de. **Avaliação e Emparcimento de Dados em um Debate de Teses**. In: Memórias del XX Congresso Internacional de Informática Educativa. 2015. .

FERNANDES JUNIOR, José I. C. **Um ambiente computacional para Arquitetura Pedagógica Debate de Teses**. 2015. Dissertação de Mestrado, Universidade Federal do Espírito Santo, Vitória.

FONSECA, Erick R.; ROSA, João L. G. Mac-morpho revisited: Towards robust part-of-speech tagging. In: **Proceedings of the 9th Brazilian Symposium in Information and Human Language Technology**. 2013. p. 98-107.

GUEDES, Gilleanes T. A. **Um Metamodelo UML para a Modelagem de Requisitos em Projetos de Sistemas MultiAgentes**. 2012. Tese de Doutorado, Universidade Federal do Rio Grande do Sul, Porto Alegre.

LANDOLI, Luca et al. Socially augmented argumentation tools: Rationale, design and evaluation of a Debate Dashboard. **International Journal of Human-Computer Studies**, v. 72, n. 3, p. 298-319, 2014.

JENNINGS, Nicholas R. On agent-based software engineering. **Artificial Intelligence**, v. 117, n. 2, p. 277-296, 2000.

KAKKONEN, Tuomo; SUTINEN, Erkki. Automatic assessment of the content of essays based on course materials. In: **Information Technology: Research and Education, 2004. ITRE 2004. 2nd International Conference on**. IEEE, 2004. p. 126-130.

KLEIN, Richard; KYRILOV, Angelo; TOKMAN, Mayya. Automated assessment of short free-text responses in computer science using latent semantic analysis. In: **Proceedings of the 16th annual joint conference on innovation and technology in computer science education**. ACM, 2011. p. 158-162.

KIM, Jihie; SHAW, Erin. Scaffolding student online discussions using past discussions: PedaBot studies. **Artificial Intelligence Review**, v. 41, n. 1, p. 97-112, 2014.

MAGALHÃES NETTO, José F. de. **Uma Arquitetura para Ambientes Virtuais de Convivência: uma Proposta Baseada em Sistemas Multiagente**. 2006. Tese de Doutorado, Universidade Federal do Espírito Santo. Vitória.

MANNING, C. D.; RAGHAVAN, P.; SCHÜTZE, H. **An Introduction to Information Retrieval**. Cambridge University Press. Edição Online. 2009. Disponível em: <<http://www.informationretrieval.org/>>. Acesso em: 20 out. 2015

MICHELS, Ana Beatriz. **Do Fazer ao Compreender no Contexto da Educação a Distância: Uso de Arquiteturas Pedagógicas no Processo de Empreender**. 2014. Dissertação de Mestrado, Universidade Federal do Rio Grande do Sul, Porto Alegre.

NEVADO, Rosane A.; MENEZES, Crediné S. de; VIEIRA JUNIOR, Ramon R. M. Debate de Teses - Uma Arquitetura Pedagógica. In: **Anais do Simpósio Brasileiro de Informática na Educação**. 2011.

ORENGO, Viviane. M.; HUYCK, Christian. A stemming algorithm for the portuguese language. In: **String Processing and Information Retrieval – SPIRE**. IEEE, 2001. p. 0186.

PÉREZ, Francisco G.; CASTILLO, Daniel P. **La mediación pedagógica**. Edição Ciccus. Buenos Aires. 1999.

PYTHON 2.7.6. 2014. Disponível em: <<https://docs.python.org/2.7/about.html>>. Acesso em: 10 jan. 2016.

RAMOZZI-CHIAROTTINO, Zélia. **Psicologia e epistemologia genética de Jean Piaget**. Epu, 1988.

REIS, Luís Paulo. **Coordenação em Sistemas Multi-Agente: Aplicações na Gestão Universitária e Futebol Robótico**. 2003. Tese de Doutorado, Faculdade de Engenharia da Universidade do Porto. Portugal.

SALTON, Gerard; WONG, Anita; YANG, Chung-Shu. A vector space model for automatic indexing. **Communications of the ACM**, v. 18, n. 11, p. 613-620, 1975. Disponível em: <<http://doi.acm.org/10.1145/361219.361220>>. Acesso em: 10 abr. 2015.

SILVA, Júlia K. K. da. et al. Automatização do processo de identificação de presença social em fóruns e chats. In: **Anais do Simpósio Brasileiro de Informática na Educação**. 2012.

STERBINI, Andrea; TEMPERINI, Marco. OpenAnswer, a framework to support teacher's management of open answers through peer assessment. In: **Frontiers in Education Conference**, 2013 IEEE. IEEE, 2013. p. 164-170.

SISTEMA DEBATE DE TESES. Disponível em: <<http://www.pead.faced.ufrgs.br/sites/cms/debate/>>. Acesso em: 17 fev. 2016.

TEP2.0 BETA. Disponível em: <<http://143.107.183.175:21480/tep2/ajuda.htm#oque>>. Acesso em: 12 dez. 2015.

WOOLDRIDGE, Michael. **An Introduction To Multiagent Systems**. John Wiley & Sons, 2009.