

UNIVERSIDADE REGIONAL DE BLUMENAU
CENTRO DE CIÊNCIAS EXATAS E NATURAIS
CURSO DE CIÊNCIA DA COMPUTAÇÃO – BACHARELADO

FERRAMENTA DE AUXÍLIO ACADÊMICO UTILIZANDO
CHATTERBOT

CAMILA VIVIANI MARTINS

BLUMENAU
2016

CAMILA VIVIANI MARTINS

**FERRAMENTA DE AUXÍLIO ACADÊMICO UTILIZANDO
CHATTERBOT**

Trabalho de Conclusão de Curso apresentado ao curso de graduação em Ciência da Computação do Centro de Ciências Exatas e Naturais da Universidade Regional de Blumenau como requisito parcial para a obtenção do grau de Bacharel em Ciência da Computação.

Profª. Joyce Martins, Mestre – Orientadora

**BLUMENAU
2016**

FERRAMENTA DE AUXÍLIO ACADÊMICO UTILIZANDO CHATTERBOT

Por

CAMILA VIVIANI MARTINS

Trabalho de Conclusão de Curso aprovado
para obtenção dos créditos na disciplina de
Trabalho de Conclusão de Curso II pela banca
examinadora formada por:

Presidente: _____
Prof^ª. Joyce Martins, Mestre – Orientadora, FURB

Membro: _____
Prof. Alexander Roberto Valdameri, Mestre – FURB

Membro: _____
Prof. Andreza Sartori, Doutora – FURB

Blumenau, 09 de dezembro de 2016

Dedico este trabalho aos meus pais, que sempre sonharam e incentivaram a conclusão do meu curso de graduação.

AGRADECIMENTOS

Agradeço primeiramente a meus pais, que sempre me apoiaram e incentivaram meus estudos.

Ao meu namorado Lucas, por todo apoio e carinho nas horas mais difíceis.

À minha orientadora Joyce, que com sua paciência, competência, dedicação e calma, me guiou desde o início.

Ao professor Valdameri, que foi a pessoa que me deu a sugestão de trabalhar com o tema *chatbot*.

À minha prima Suzane, que tentou me passar um pouco do seu conhecimento em web.

Ao meu grande amigo Rafael, que me acompanhou durante alguns sábados de estudos na biblioteca.

À minha família, amigos e à família do meu namorado, por toda compreensão neste momento.

A todos professores e colegas de trabalho, que compartilharam seus conhecimentos durante os quatro anos de graduação.

E aos meus amigos e colegas de curso: Luís, Jean, Thainara, Filipe, Philip, Daniel e Juliano, que tornaram esses quatro anos de estudos mais agradáveis e divertidos.

Faça o melhor que puder. Seja o melhor que puder. O resultado virá na mesma proporção de seu esforço.

Mahatma Gandhi

RESUMO

Este trabalho apresenta o desenvolvimento de um *chatterbot* para auxílio acadêmico, que é capaz de manter uma conversação em língua portuguesa, respondendo ou fazendo perguntas sobre um texto da educação básica na área de conhecimento em Geografia. A partir do texto previamente inserido pelo aluno, a aplicação gera perguntas e respostas que são gravadas em uma base de conhecimento Artificial Intelligence Markup Language (AIML). Para tanto, utiliza-se o *parser* Palavras para efetuar a análise morfológica e obter os papéis semânticos de cada palavra em cada sentença que compõe o texto. Dentre os papéis semânticos identificados pelo *parser* Palavras, foram tratados sete, entre eles agente, localização e extensão, em sentenças composta por sujeito, verbo e predicado. O ponto principal para a geração das perguntas é o papel semântico, a partir do qual é possível definir o pronome ou o advérbio interrogativo, tais como quem, quando, quanto, qual e onde, que deve ser usado para elaborar a pergunta. Além disso, são feitos ajustes para possibilitar a conjugação verbal em determinados casos e criar perguntas mais concisas. O trabalho proposto atingiu os objetivos elencados, com limitações devido à complexidade da língua portuguesa, mostrando-se relevante como uma ferramenta de processamento de linguagem natural, bem como uma aplicação auxiliar para o processo de ensino-aprendizagem.

Palavras-chave: *Chatterbot*. Papéis semânticos. Geração de bases de conhecimento AIML.

ABSTRACT

This monograph presents the development of a chatterbot for academic assistance, which is able to maintain a conversation in Portuguese language, answering or asking questions about a basic education text on the knowledge area of Geography. From the text previously entered by the student, the application generates questions and answers that are written in an Artificial Intelligence Markup Language (AIML) knowledge base. To do so, the parser is used to perform the morphological analysis and to obtain the semantic roles of each word in each sentence that composes the text. Seven semantic roles identified by the parser Palavras were treated, among them agent, localization and extension, in sentences composed by subject, verb and predicate. The main point for generating the questions is the semantic role, from which it is possible to define the interrogative pronoun or adverb, such as who, when, how much, what and where, that should be used to elaborate the question. In addition, adjustments are made to allow verbal conjugation in certain cases and to create more concise questions. The proposed work reached the objectives listed, with limitations due to the complexity of the Portuguese language, and it is relevant as a natural language processing tool, as well as an auxiliary application for the teaching-learning process.

Key-words: *Chatterbot*. Semantic Roles. Creation of AIML knowledge bases.

LISTA DE FIGURAS

Figura 1 – Arquitetura do <i>chatbot</i> ALICE	15
Figura 2 – Análise <i>full morphosyntactic parse</i> da frase “João abraçou a sua avó.”	19
Figura 3 – Análise <i>semantic roles</i> da frase “João abraçou a sua avó.”	19
Figura 4 – Arquitetura geral de um sistema de perguntas e respostas.....	20
Figura 5 – Exemplo de interação com a professora Elecktra	22
Figura 6 – Exemplo de interação com o ChatterDóris	24
Figura 7 – Arquitetura do ChatterEdu	26
Figura 8 – Análise morfológica da frase "A Ásia é o continente mais populoso"	27
Figura 9 – Diagrama de casos de uso	30
Figura 10 – Diagrama de classes	31
Figura 11 – Tela inicial do protótipo	36
Figura 12 – Interação com o ChatterEdu: usuário faz pergunta ao robô.....	37
Figura 13 – Interação com o ChatterEdu: usuário responde uma pergunta de forma correta ..	37
Figura 14 – Interação com o ChatterEdu: usuário responde uma pergunta de forma errada ...	38

LISTA DE QUADROS

Quadro 1 – Trecho de código AIML	16
Quadro 2 – Trecho de código AIML usando <random>.....	17
Quadro 3 – Requisitos funcionais.....	25
Quadro 4 – Requisitos não funcionais.....	25
Quadro 5 – Papéis semânticos e suas respectivas interrogativas	29
Quadro 6 – Método perguntaLOC da classe PerguntasERespostas	34
Quadro 7 – Comparativo entre o ChatterEdu e os trabalhos correlatos	40
Quadro 8 – Caso de uso UC01: Inserir texto.....	47
Quadro 9 – Caso de uso UC02: Dialogar com o ChatterEdu.....	47
Quadro 10 – Base de conhecimento gerada.....	48

LISTA DE ABREVIATURAS E SIGLAS

AIML – Artificial Intelligence Markup Language

ALICE – Artificial Linguistic Internet Computer Entity

HTML – HyperText Markup Language

HTTP - HyperText Transfer Protocol

LSI – Latent Semantic Indexing

PLN – Processamento de Linguagem Natural

RF – Requisitos Funcionais

RNF – Requisitos Não Funcionais

UC – Use Case

SUMÁRIO

1 INTRODUÇÃO.....	12
1.1 OBJETIVOS.....	13
1.2 ESTRUTURA.....	13
2 FUNDAMENTAÇÃO TEÓRICA	14
2.1 CHATTERBOTS	14
2.2 LINGUAGEM AIML.....	15
2.3 PAPÉIS TEMÁTICOS	17
2.4 GERAÇÃO AUTOMÁTICA DE PERGUNTAS	19
2.5 TRABALHOS CORRELATOS	20
2.5.1 Elektra	21
2.5.2 Tuxbot	22
2.5.3 ChatterDóris	23
3 DESENVOLVIMENTO DO CHATTEREDU.....	25
3.1 REQUISITOS.....	25
3.2 ESPECIFICAÇÃO	26
3.2.1 Arquitetura	26
3.2.2 Análise morfológica do site Palavras.....	27
3.2.3 Casos de uso.....	29
3.2.4 Diagramas de classes.....	30
3.3 IMPLEMENTAÇÃO	32
3.3.1 Técnicas e ferramentas utilizadas.....	32
3.3.2 Processamento do texto de entrada	32
3.3.3 Operacionalidade da implementação	35
3.4 ANÁLISE DOS RESULTADOS	38
4 CONCLUSÕES.....	42
4.1 EXTENSÕES	43
REFERÊNCIAS	45
APÊNDICE A – DETALHAMENTO DOS CASOS DE USO	47
APÊNDICE B – DETALHAMENTO DAS BASES DE CONHECIMENTO AIML	48

1 INTRODUÇÃO

Atualmente as novas tecnologias têm proporcionado rápidas e profundas mudanças em diversos setores da sociedade. Este processo de evolução também é percebido no meio acadêmico, afetando não apenas alunos, mas também professores, coordenadores pedagógicos e diretores (BARBOSA, 2012). Como consequência, percebe-se que o uso de ferramentas computacionais tem se tornado cada vez mais comum no auxílio de alunos e professores em suas atividades cotidianas.

Dentre diversos softwares de ensino disponíveis atualmente, é possível encontrar sistemas que utilizam *chatbots* para comunicar-se com o usuário a fim de lhe proporcionar o aprendizado de um assunto específico a partir da iteração com o aplicativo. Segundo Comarella e Café (2008), os *chatbots* são sistemas computacionais que simulam a conversação de um ser humano. Desta forma, um dos intuitos de utilizar um sistema como o supracitado para auxílio acadêmico seria o de proporcionar para o aluno uma experiência semelhante a que teria numa conversa on-line com um professor ou especialista de uma determinada área.

Neste contexto, um dos alicerces necessários para o desenvolvimento de tal tecnologia são as técnicas de Processamento de Linguagem Natural (PLN), que é uma subárea da Inteligência Artificial. Comarella e Café (2008) afirmam que o PLN utiliza de conhecimentos em linguística e comunicação para melhorar a interação entre sistemas computacionais e seres humanos, podendo ser simplificado como a habilidade de um computador em processar a mesma linguagem que as pessoas usam no seu dia-a-dia, ou seja, a linguagem natural.

Considerando o atual cenário relacionado à área de PLN e ao desenvolvimento de *chatbots*, surgem questionamentos que vêm sendo estudados por pesquisadores ao redor do mundo. Uma das inquietações que se pretende elucidar a partir do projeto descrito no presente documento é: como desenvolver um aplicativo que utilize a tecnologia de um *chatbot* para manter uma conversação em português baseada em perguntas e respostas sobre um texto acadêmico previamente escolhido e inserido pelo usuário?

Com base no exposto, foi desenvolvida uma ferramenta capaz de receber como entrada do usuário um texto com conhecimentos da educação básica, a partir do qual são formuladas perguntas e respostas. O insumo obtido a partir do processamento realizado no texto de entrada é transformado em uma base de conhecimento no formato conhecido como Artificial

Intelligence Markup Language (AIML), que é utilizada para realizar a conversação entre o *chatterbot* e o aluno.

1.1 OBJETIVOS

Este trabalho visa desenvolver uma ferramenta de auxílio que utilize *chatterbot* para possibilitar uma conversação com o usuário sobre textos acadêmicos.

Os objetivos específicos do trabalho são:

- a) permitir interação em língua portuguesa;
- b) possibilitar a interação entre o usuário e o *chatterbot* através de uma página web;
- c) processar textos da educação básica na área de conhecimento em Geografia, gerando perguntas e respostas;
- d) possibilitar que o usuário faça perguntas ao *chatterbot* ou responda aos questionamentos feitos acerca do texto escolhido.

1.2 ESTRUTURA

A presente monografia encontra-se dividida em quatro capítulos: introdução, fundamentação teórica, desenvolvimento do ChatterEdu e conclusões. O capítulo 2 visa fornecer um embasamento teórico a respeito dos principais assuntos abordados no presente trabalho, explanando conceitos como *chatterbots*, linguagem AIML, papéis temáticos e o *parser* Palavras. Também versa sobre o conteúdo referente à geração automática de perguntas e respostas e, por fim, apresenta três trabalhos correlatos a este. Na sequência, o capítulo 3 expõe os principais pontos do desenvolvimento do protótipo como: os requisitos, a arquitetura, o processamento do retorno obtido do site Palavras e os diagramas de casos de usos e de classes. Além disso, também são apresentadas as ferramentas utilizadas, a implementação e operacionalidade da aplicação, bem como são analisados os resultados obtidos. O capítulo 4 encerra a monografia expondo as conclusões obtidas no presente trabalho e sugere extensões que poderão ser feitas a partir deste.

2 FUNDAMENTAÇÃO TEÓRICA

Este capítulo tem como objetivo explorar os principais assuntos necessários para a realização do trabalho. Desta forma, o capítulo foi subdividido em cinco partes, onde a seção 2.1 conceitua *chatbots*, mostra suas três gerações e descreve sua arquitetura. Já a seção 2.2 apresenta a linguagem AIML. A seção 2.3 apresenta os papéis semânticos e a ferramenta de análise morfológica Palavras, utilizada para identificá-los e, assim, permitir a elaboração de perguntas. Na seção 2.4 é apresentada uma explanação sobre a geração automática de perguntas. Por fim, na seção 2.5 são descritos três trabalhos correlatos.

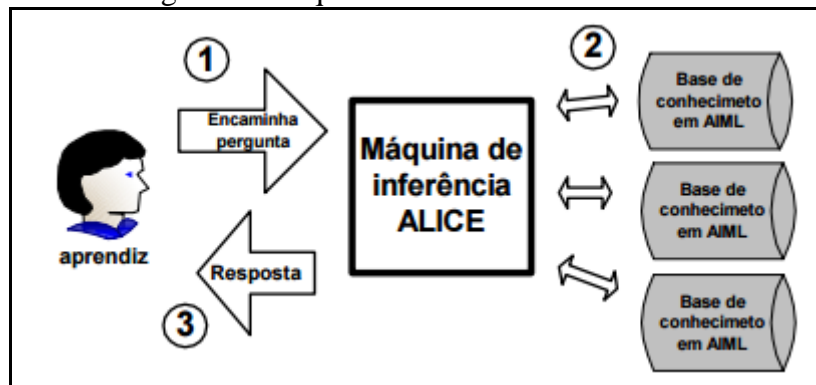
2.1 CHATTERBOTS

Segundo Bickmore (1999 apud GALVÃO, 2003, p. 18) e Laven (2002 apud GALVÃO, 2003, p. 18), “*chatbots* são sistemas que utilizam linguagem natural para dialogar com o usuário”. Com base neste conceito, esta tecnologia objetiva ludibriar o usuário a fim de que o mesmo acredite estar conversando com outro ser humano e não com um computador. A partir desta premissa, é possível imaginar a aplicabilidade dos *chatbots* nas mais diversas áreas.

Ainda citando Galvão (2003), no princípio os *chatbots* tinham sua aplicação restrita a estudos acadêmicos. Entretanto, com o passar do tempo, eles estão sendo inseridos em diferentes segmentos, como, por exemplo, na área de entretenimento, comércio eletrônico, atendimento ao consumidor e ensino à distância. Tendo em vista a evolução percebida na área de *chatbots*, Moura (2003) faz um breve levantamento histórico sobre o tema, onde divide os robôs de conversação em três gerações:

- a) 1ª geração: o marco desta geração foi o pioneiro *chatbot* conhecido como Eliza, projetada para dialogar com as pessoas fazendo o papel de um psicanalista, apenas utilizando regras gramaticais e casamento de padrões;
- b) 2ª geração: esta geração ficou marcada por introduzir técnicas e conceitos da Inteligência Artificial para o desenvolvimento dos robôs de conversação, como aprendizado de máquina, redes neurais e lógica fuzzy;
- c) 3ª geração: neste momento surge o sistema conhecido como Artificial Linguistic Internet Computer Entity (ALICE), que utiliza a linguagem de marcação AIML para representar as bases de conhecimento do *chatbot*.

Para o melhor entendimento da última geração elaborada, é necessário entender como funciona a arquitetura de um *chatbot* baseado no projeto ALICE, descrita por Teixeira (2005) e ilustrada na Figura 1.

Figura 1 – Arquitetura do *chatterbot* ALICE

Fonte: Teixeira (2005, p. 26).

No primeiro passo, o usuário entra com a pergunta, que terá que passar por uma série de processamentos linguísticos para transformá-la numa linguagem apropriada para realizar a busca nas bases de conhecimento. Este passo fará a simplificação e normalização da pergunta de entrada.

Depois disto, é necessário fazer uma padronização das sentenças. Então serão mantidas apenas as palavras-chave e o restante do texto será descartado. Além disso, é usado o asterisco (*) para representar quando uma palavra-chave pode ser precedida ou seguida de uma ou mais palavras, como, por exemplo, a frase: “Por favor me passe a xícara de chá.” pode ser escrita como “* passe a xícara de chá.”, já que as outras palavras não são essenciais para o entendimento da mesma.

A partir do momento que a frase de entrada esteja devidamente padronizada, será possível fazer a busca nas bases de conhecimento AIML implementadas. Conforme o exposto por Teixeira (2005), as bases de conhecimento devem ser carregadas em ordem alfabética. Este tratamento tem a finalidade de evitar problemas de busca e auxiliar na manutenção da base. Por fim, se a pergunta de entrada devidamente padronizada for encontrada dentro da base de conhecimento, o *chatterbot* irá fornecer uma resposta ao usuário.

2.2 LINGUAGEM AIML

O surgimento da linguagem AIML se deu no âmbito do desenvolvimento do projeto ALICE. A AIML é uma linguagem de marcação baseada na eXtensible Markup Language (XML) e tem a finalidade de especificar as bases de conhecimentos utilizadas em *chatterbots* (WALLACE, 2000 apud OLIVEIRA, 2015).

Pilastri e Brega (2009) afirmam que essa linguagem facilita na criação de robôs de conversação e que é baseada em um conjunto de *tags* e comandos para a criação das bases de conhecimento. Neste contexto, os autores expõem algumas *tags* básicas da linguagem:

- a) `<aiml>`: define o início e o fim de um bloco de comandos da base de conhecimento;
- b) `<category>`: identifica uma unidade de conhecimento, ou seja, é usada para definir as possíveis perguntas e respostas da conversação;
- c) `<pattern>`: define o padrão de uma possível mensagem digitada pelo usuário;
- d) `<template>`: indica a resposta que o *chatterbot* dará para um determinado `<pattern>` de entrada;
- e) `<random>`: usada para selecionar respostas aleatórias;
- f) ``: utilizada em conjunto com a *tag* `<random>` para inserir as opções de respostas disponíveis;
- g) `<that>`: para dar sequência ao diálogo, essa *tag* permite registrar a última sentença;
- h) `<srai>`: redireciona para outra categoria ou questão;
- i) `<javascript>`: executa um comando em JavaScript.

Para um melhor entendimento do funcionamento de algumas das *tags* supracitadas, seguem nos Quadro 1 e Quadro 2 exemplos de trecho de código em AIML.

Quadro 1 – Trecho de código AIML

```

<aiml>
  <category>
    <pattern>Olá</pattern>
    <template>Oi, como vai você?</template>
  </category>
  <category>
    <pattern>Oi</pattern>
    <template><srai>Olá</srai></template>
  </category>
</aiml>

```

Fonte: Oliveira (2010).

O código mostrado acima utiliza as principais *tags* para representar uma conversação entre o *chatterbot* e o usuário. Na primeira *tag* `<category>` é implementado o conhecimento para quando o usuário digitar “Olá” o sistema responder “Oi, como vai você?”. Na segunda *tag* `<category>` é usado o recurso `<srai>`, que permite fazer um redirecionamento para quando o usuário disser “Oi” ser levado para o mesmo `<template>` que responde a “Olá”, já que são entradas similares que podem receber a mesma resposta do sistema. Assim, tanto para “Oi” quanto para “Olá”, o usuário receberá a resposta “Oi, como vai você?”.

Quadro 2 – Trecho de código AIML usando <random>

```

<aiml>
  <category>
    <pattern>* TUDO BEM</pattern>
    <template>
      <random>
        <li>Tudo em ordem.</li>
        <li>Muito bem, obrigado.</li>
        <li>Eu estou bem.</li>
        <li>Estou sim.</li>
        <li>Muito bem e você?</li>
        <li>Tudo legal.</li>
      </random>
    </template>
  </category>
</aiml>

```

Fonte: Oliveira (2010).

No Quadro 2 é usado o asterisco (*) para indicar para o interpretador AIML que o usuário pode digitar uma ou mais palavras antes da sentença “tudo bem”. Depois disto, tem-se a tag <random>, que utiliza a tag para incluir as possíveis respostas. Desta forma, durante a execução do código acima, se o usuário entrar, por exemplo, com a sentença “querido *chatterbot*, tudo bem?”, ele poderá receber qualquer uma das respostas incluídas em <random>, escolhida de forma aleatória pelo interpretador. Essas duas tags podem ser utilizadas para dar uma maior naturalidade na conversação, já que em momentos diferentes o usuário receberá diferentes respostas para a mesma entrada. Já no âmbito dos *chatterbots* educacionais, podem servir para dar uma segunda opção de resposta caso a primeira explicação dada pelo aplicativo não esclareça a dúvida do aluno.

2.3 PAPÉIS TEMÁTICOS

Segundo Soares e Menuzzi (2010, p. 15), “Para uma ideia inicial, pode-se dizer que papéis temáticos são relações que estão diretamente ligadas ao significado dos verbos.”. Os autores ressaltam ainda que os papéis temáticos podem ser chamados de diferentes formas, como papéis semânticos, relações temáticas, entre outras nomenclaturas. Já Kipper (2005 apud SCARTON, 2013) conceitua papéis temáticos como a relação dos significados que ficam implícitos entre o verbo e seus argumentos em uma oração, sendo utilizados para verificar padrões no comportamento do verbo. Acerca disso, para tornar os conceitos supracitados mais compreensíveis, avalia-se a sentença “João abraçou sua avó.”. Os papéis semânticos que poderiam ser definidos nessa oração são: (João) agente da oração; (abraçou) verbo; (avó) paciente da oração.

Kipper (2005 apud SCARTON, 2013) definiu vinte e três papéis temáticos para determinar as relações semânticas dos argumentos dos verbos. Alguns desses papéis definidos são relacionados a seguir:

- a) agente: argumento da oração (quem ou por quem) que executa uma ação ou evento, como em: João (agente) abraçou sua avó (paciente);
- b) destino: argumento que representa o final de um movimento, por exemplo: Eu (agente) chutei a bola até o gol (destino);
- c) fonte: argumento que indica o ponto inicial de um movimento, como na frase: Eu (agente) viajei de Navegantes (fonte) até São Paulo (destino);
- d) paciente: argumento que sofre uma ação ou é submetido a um evento, como em: João (agente) abraçou sua avó (paciente);
- e) posição: argumento que expressa uma posição, como na sentença: Santa Catarina fica na região sul (posição);
- f) tema: argumento que está localizado em um lugar ou se move de um lugar para outro, assim como ocorre na seguinte oração: a Ásia (tema) é o continente mais populoso.

No entanto, tanto Kipper (2005 apud SCARTON, 2013, p. 16) quanto Cançado (2005, apud SCARTON, 2013, p. 16) afirmam que “não existe um consenso sobre o conjunto de papéis necessários para caracterizar os argumentos de um verbo e não há nenhum critério claro para determinar quais papéis devem ser aplicados a um argumento em particular [...]”. Desta forma, quem pretende trabalhar com papéis semânticos é incumbido de definir quais classificações semânticas serão adotadas.

Para determinar os papéis temáticos podem ser utilizadas ferramentas de anotação para fazer o processamento do texto de entrada, como o *parser* Palavras (BICK et al., 2016), componente do projeto Visual Interactive Syntax Learning (VISL). Na documentação dessa ferramenta existe uma lista com mais de cinquenta papéis temáticos definidos. Além disso, é possível realizar cinco tipos de análises diferentes para um texto de entrada, dentre os quais cita-se a opção *full morphosyntactic parse*, para efetuar um processamento morfossintático, e a opção *semantic roles*, para determinar os papéis semânticos. Assim, na Figura 2 pode-se visualizar o processamento morfossintático da frase “João abraçou sua avó.”, enquanto a Figura 3 traz os papéis semânticos dessa mesma oração.

Figura 2 – Análise *full morphosyntactic parse* da frase “João abraçou a sua avó.”

Enter Portuguese text to parse:
João abraçou sua avó

Parser: Full morphosyntactic parse Visualization: Default

Go! Reset

João [João] <hum> **PROP M S @SUBJ>**
abraçou [abraçar] <fmc> <vt> **V PS 3S IND VFIN @FMV**
sua [seu] <poss 3S> <si> **DET F S @>N**
avó [avó] <Hfam> **N F S @<ACC**

Fonte: Bick et al. (2016).

Figura 3 – Análise *semantic roles* da frase “João abraçou a sua avó.”

Enter Portuguese text to parse:
João abraçou sua avó

Parser: Semantic roles Visualization: Default

Go! Reset

<β>
João [João] **PROP M S @SUBJ>** §AG #1->2
abraçou [abraçar] <fmc> **V PS 3S IND VFIN @FS-STA** §PRED #2->0
sua [seu] <poss 3S> <si> **DET F S @>N** #3->4
avó [avó] <Hfam> **N F S @<ACC** §PAT #4->2 </β>

Fonte: Bick et al. (2016).

Nas Figura 2 e Figura 3 tem-se o resultado do processamento do texto de entrada, onde em cada linha encontra-se a palavra tal como escrita na frase, seguida da palavra na sua forma base entre colchetes. Algumas palavras trazem logo em seguida uma *tag* entre os símbolos maior e menor, que é uma classificação secundária da palavra. Nos exemplos, <hum> significa pessoa ou indivíduo, enquanto <fmc> significa que a palavra é o verbo principal da oração. Em azul e negrito tem-se a classe gramatical da palavra e o que não está em negrito representa as flexões possíveis (gênero, número, grau) ou os tempos verbais, quando tratar-se de um verbo. As *tags* em verde definem os termos sintáticos que aquela palavra assume dentro da oração. Assim, por exemplo, @SUBJ> significa que é o sujeito, enquanto @FS-STA é o verbo principal. Já a *tag* precedida do símbolo § representa o papel semântico da palavra. Na Figura 3, §AG é o agente, §PRED é o verbo principal com os quais os demais argumentos se relacionam, §PAT é o paciente.

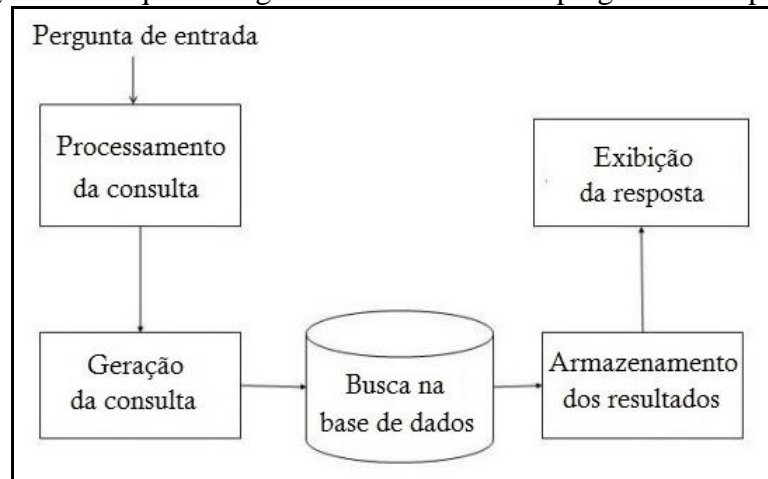
2.4 GERAÇÃO AUTOMÁTICA DE PERGUNTAS

Rus et al. (2008 apud LE; KOJIRI; PINKWART, 2014) definem a geração de perguntas como a tarefa de receber uma entrada qualquer, como um texto, uma base de dados ou uma base semântica, e a partir destas gerar perguntas automaticamente. Os autores ainda afirmam que as pesquisas em torno da geração automática de perguntas já têm sido

conduzidas há algum tempo, porém somente nos últimos anos essas pesquisas voltaram-se para estudos de geração automáticas de perguntas aplicados na educação, que é o caso do presente trabalho.

Além disso, Gupta e Gupta (2012) afirmam que as técnicas utilizadas para formar perguntas e respostas são multidisciplinares, pois abrangem Inteligência Artificial, processamento de linguagem natural, ciência cognitiva e o conhecimento em gerenciamento de bases de dados. Também atestam que os sistemas de perguntas e respostas costumam apresentar três etapas distintas: a classificação da pergunta, o processamento do documento ou informação inserida e a extração da resposta. Ainda segundo Gupta e Gupta (2012), a arquitetura geral de um sistema de geração de perguntas e respostas segue o modelo apresentado na Figura 4.

Figura 4 – Arquitetura geral de um sistema de perguntas e respostas



Fonte: adaptado de Gupta e Gupta (2012, p. 2).

Nessa arquitetura, o sistema recebe uma pergunta como entrada do usuário e parte para o processamento desta, onde o sistema verifica se a questão recebida se encaixa em algum dos tipos suportados. Após essa etapa, é montada uma consulta para ser feita na base de dados ou na base de conhecimento da aplicação. Os resultados da base de dados são obtidos a partir de uso de regras e palavras-chaves e são gravados em um arquivo, para só no passo seguinte serem exibidos para o usuário como resposta da pergunta de entrada.

2.5 TRABALHOS CORRELATOS

Foram selecionados três trabalhos correlatos ao proposto, todos são projetos de *chatterbots* utilizados para a educação. Primeiramente, é apresentado o robô de conversação chamado de professora Elektra (LEONHARDT et al., 2003). Em seguida, é exposto o

conteúdo do projeto Tuxbot (TEIXEIRA, 2005). Por fim, é comentado sobre o desenvolvimento do ChatterDóris (KÜHLEIS et al., 2012).

2.5.1 Elektra

A professora Elektra foi desenvolvida na Universidade Federal do Rio Grande do Sul (UFRGS), em 2002, com base no *chatbot* ALICE. Segundo Leonhardt et al. (2003), inicialmente a Elektra foi feita com o intuito de responder perguntas sobre física para alunos do ensino médio que estivessem em fase preparatória para o vestibular. No ano de 2003 este projeto foi estendido para atender também os alunos de um curso de educação à distância em informática na educação, aumentando sua base de conhecimento para abranger também conteúdos relacionados a redes de computadores e internet.

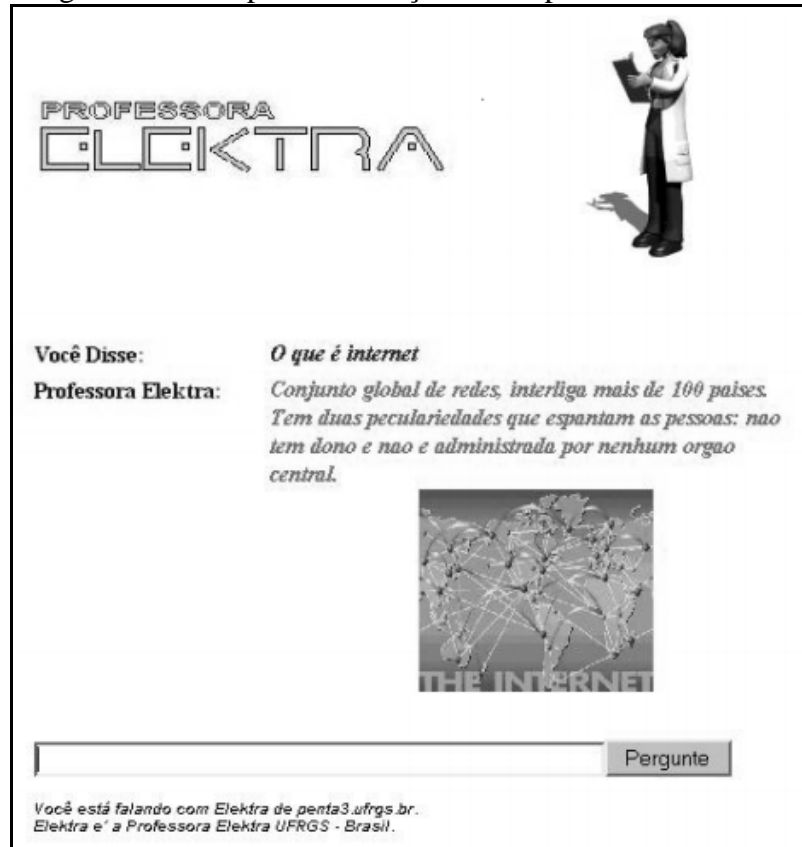
Para tornar o diálogo entre o sistema e os usuários mais natural, foram utilizadas diversas funcionalidades da linguagem AIML. Uma delas foi a de adicionar mais de uma resposta para uma mesma pergunta de entrada, sendo escolhida de forma randômica, permitindo ao usuário a opção de perguntar sobre o mesmo conteúdo e receber respostas diferentes caso a primeira explicação não tenha sanado todas as suas dúvidas. Para implementar tal funcionalidade, foram utilizadas as *tags* `<random>` e ``, já explicadas anteriormente. Além disso, os desenvolvedores optaram por disponibilizar *hyperlinks* e imagens para melhor esclarecimento das questões abordadas, como pode ser observado na Figura 5.

Os autores citam que também foram utilizadas outras *tags* para simplificar a implementação e tornar o *chatbot* mais eficiente. Entretanto, foram recuperados os *logs* das conversas para avaliar se a conversação estava ocorrendo de forma natural e nesta análise percebeu-se que o sistema Elektra trazia informações muito ricas em seu conteúdo, mas a comunicação não estava adequada. Foi constatado que as respostas do *chatbot* eram muito diretas e objetivas, desestimulando o diálogo. Outro problema percebido foi que a variação da linguagem utilizada em diversas regiões do Brasil dificultava o trabalho de reconhecimento da pergunta feita pelo usuário, deixando de trazer respostas que estavam em sua base de conhecimento por conta do não entendimento da pergunta efetuada. Partindo destas constatações, foram feitos ajustes para deixar o *chatbot* mais amigável e o usuário mais interessado na conversação, além de melhorar essa compreensão de vocabulários de diferentes regiões.

Por fim, os desenvolvedores do projeto Elektra acreditam que os *chatbots* educacionais despertam a curiosidade e motivam os alunos por ser uma forma diferente de

aprender os conteúdos acadêmicos. Mas também apontam que a professora Elektra necessita de uma maior utilização e maiores análises dos *logs* para deixar a conversação mais natural e abrangente.

Figura 5 – Exemplo de interação com a professora Elektra



Fonte: Leonhardt et al. (2003).

2.5.2 Tuxbot

O trabalho elaborado por Teixeira (2005) visa desenvolver um *chatbot* que seja capaz de alimentar sua própria base de conhecimento. Neste contexto, o objetivo do projeto é facilitar a busca de informações no site FocaLinux. Para realizar tal tarefa, Teixeira (2005) explica que a estratégia que decidiu adotar foi a modelagem do conhecimento através da recuperação de padrões de conversação baseado na linguística de corpus, na categorização desses padrões usando a metodologia Latent Semantic Indexing (LSI) e na conversão destes padrões em conhecimento para o *chatbot*. A partir dessas etapas, foi desenvolvido o *chatbot* Tuxbot, para esclarecimento de dúvidas sobre Linux. Um ponto importante que é abordado é a definição de corpus, que é um conjunto de texto de determinado assunto em linguagem natural feito especificamente para pesquisas linguísticas e, por sua vez, corpora é o plural de corpus.

Ao longo do trabalho, o autor explica como funciona o passo-a-passo para a construção das bases de conhecimento. Primeiramente são definidos os padrões lexicais de cada palavra e montado um corpus e são estudadas as associações, estruturas e correlações dos sentidos das palavras destacadas. Para criar as bases de conhecimento em AIML são utilizados três tipos de corpora: um de perguntas genéricas, um de palavras-chave e outro de perguntas específicas. Após essa etapa, é realizada uma categorização e um tratamento utilizando a técnica LSI a fim de evitar dados repetidos e simplificar a análise do corpora. Por fim, são extraídas apenas as categorias de interesse para a base de conhecimento e são gerados pares de padrões, que serão correspondentes às perguntas e respostas da base.

Para finalizar o trabalho, foi feito um estudo de caso que utilizou aproximadamente 9Mb de arquivos de texto sobre o Linux e após a execução buscando o texto “Linux?” foi gerado um corpus com 200 padrões, divididos em 6 categorias. Desta forma, foi possível constatar que o algoritmo LSI categorizou 100% dos padrões.

2.5.3 ChatterDóris

Kühleis et al. (2012) descrevem que o objetivo principal deste trabalho é incrementar o projeto Dóris, que inicialmente era um agente pedagógico que demonstra emoções na sua interação com os estudantes. Dóris é um Sistema Tutor Inteligente (STI) que busca caracterizar o perfil dos estudantes e, a partir disto, orientá-los e acompanhá-los nas suas atividades. Já o projeto ChatterDóris veio com o intuito de evoluir o projeto Dóris para o mesmo conseguir se comunicar com os usuários usando linguagem natural, tendo o objetivo de esclarecer dúvidas sobre os conteúdos abordados nas aulas.

Para a implementação do projeto foi utilizado o Program D, um processador de linguagem natural voltado para o desenvolvimento de *chatterbots*, que usa como base de conhecimento arquivos AIML. A escolha da ferramenta supracitada deu-se por conta desta ser compatível com a tecnologia já utilizada no projeto Dóris, que foi desenvolvido em Java. A interação com o ChatterDóris ocorre a partir de uma caixa de texto, como pode ser observado na Figura 6.

Para avaliar o projeto, foi disponibilizado um protótipo para voluntários formandos do curso de Pedagogia e a métrica levada em consideração foi a satisfação dos usuários do sistema no âmbito educacional. Para julgar o desempenho do ChatterDóris foram consideradas: a eficiência e a qualidade do diálogo, a satisfação dos usuários e a avaliação na forma de um pedido informal para comentários. A partir da métrica estabelecida, percebeu-se que o *chatterbot* teve um bom desempenho, mas ainda deixou muito a desejar no quesito de

qualidade das respostas, que muitas vezes não foram coerentes com a pergunta feita pelo usuário. Entretanto, mesmo o protótipo não sendo tão eficiente quanto à conversação com um humano, ainda assim os voluntários avaliadores da proposta acreditam ser algo positivo a inserção deste tipo de agente no âmbito educacional.

Figura 6 – Exemplo de interação com o ChatterDóris



Figura 8: Dóris com o mecanismo Chatterbot implementado

Figura 9: Dóris demonstrando alegria ao usuário

Figura 10: Dóris demonstrando atenção ao usuário

Figura 21: Dóris demonstrando expectativa ao usuário

Fonte: Kühleis et al. (2012).

3 DESENVOLVIMENTO DO CHATTEREDU

O presente capítulo apresenta o desenvolvimento de um *chatbot* chamado de ChatterEdu. Inicia-se com a descrição dos requisitos funcionais e não funcionais levantados para o desenvolvimento da aplicação (na seção 3.1), seguida da especificação do trabalho (na seção 3.2), que apresenta a arquitetura do protótipo desenvolvido, bem como a análise morfológica retornada do site Palavras, além dos diagramas de casos de uso, de pacotes e de classes. Já a seção 3.3 aborda assuntos referentes à implementação, listando as ferramentas utilizadas no desenvolvimento e explicando sobre a operacionalidade da aplicação. E, por fim, a seção 3.4 analisa os resultados obtidos a partir do presente projeto.

3.1 REQUISITOS

O projeto proposto deve atender aos Requisitos Funcionais (RF) e Requisitos Não Funcionais (RNF) apresentados no Quadro 3 e Quadro 4, respectivamente. Os requisitos funcionais estão relacionados aos casos de uso apresentados na Figura 9.

Quadro 3 – Requisitos funcionais

Requisitos funcionais (RF)	Casos de uso (UC)
RF01: estabelecer a interação entre o usuário e o <i>chatbot</i> a partir de uma interface desktop	UC01, UC02
RF02: possibilitar a interação com o usuário utilizando a língua portuguesa	UC01, UC02
RF03: permitir que o usuário forneça como entrada um texto da educação básica na área de conhecimento em Geografia com sentenças composta por sujeito, verbo e predicado	UC01
RF04: alimentar a base de conhecimento a partir do processamento dos textos de entrada, gerando perguntas e respostas	UC01
RF05: permitir que o usuário faça perguntas sobre o texto ao <i>chatbot</i>	UC03
RF06: permitir que o usuário responda às perguntas feitas pelo <i>chatbot</i>	UC04
RF07: ter uma base AIML com respostas padrões a serem usadas quando o <i>chatbot</i> não conseguir responder às perguntas do usuário	UC05
RF08: ter uma base AIML com saudações padrões	UC06

Fonte: elaborado pela autora.

Quadro 4 – Requisitos não funcionais

Requisitos não funcionais (RNF)
RNF01: usar a linguagem Java para desenvolvimento do <i>chatbot</i>
RNF02: utilizar o ambiente de desenvolvimento Netbeans para a implementação do projeto
RNF03: utilizar a linguagem AIML para construção das bases de conhecimento do <i>chatbot</i>
RNF04: utilizar o Program AB para interpretar as bases de conhecimento AIML
RNF05: é necessária conexão com a internet para utilizar a aplicação
RNF06: utilizar o analisador morfológico Palavras para determinar os papéis semânticos do texto de entrada

Fonte: elaborado pela autora.

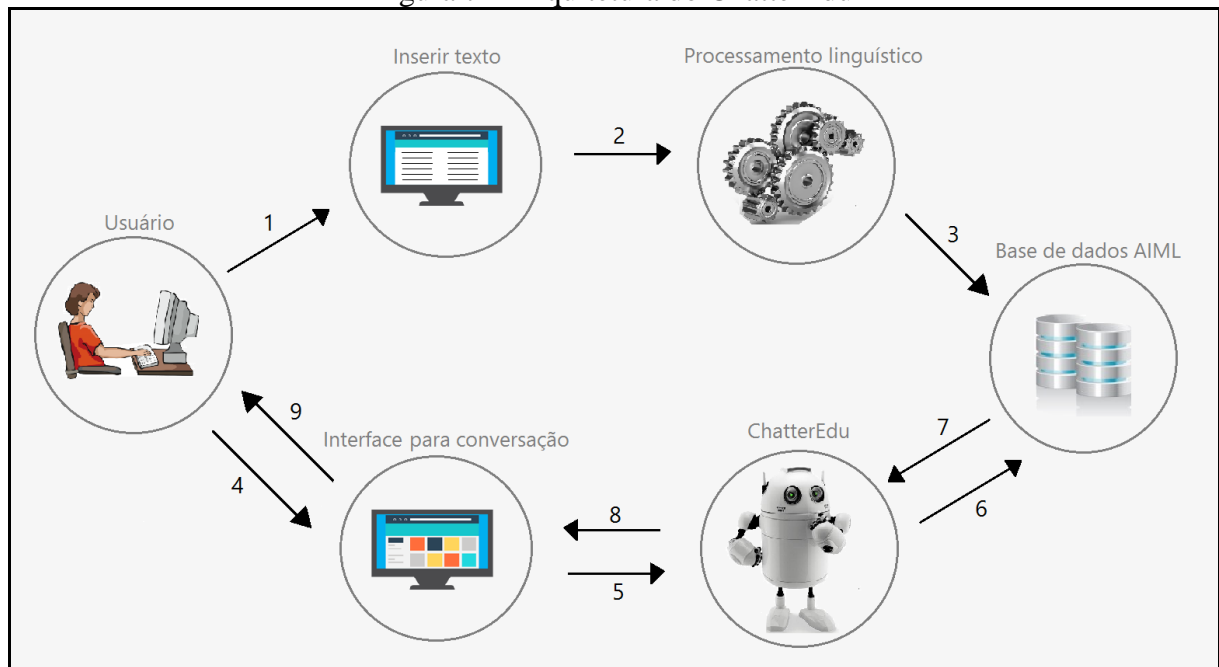
3.2 ESPECIFICAÇÃO

Nesta seção é descrita a especificação do ChatterEdu. Inicialmente é apresentada a arquitetura do protótipo, seguida da explanação de como a aplicação utiliza a análise morfológica retornada pelo site Palavras para gerar as perguntas e respostas automaticamente. Além disso, são expostos os diagramas de casos de uso, de pacotes e de classes, desenvolvidos com a ferramenta Enterprise Architect.

3.2.1 Arquitetura

A Figura 7 exibe a arquitetura do *chatterbot* desenvolvido.

Figura 7 – Arquitetura do ChatterEdu



Fonte: elaborado pela autora.

Primeiramente o usuário deve entrar com um texto referente a conhecimentos da educação básica da área de Geografia (1). O texto de entrada passa por um pré-processamento onde são obtidos o papel semântico e a classificação morfológica de cada palavra. A partir disso, é realizado mais um processamento para transformar as sentenças de entrada em perguntas e respostas (2). Essas perguntas e respostas são tratadas para serem gravadas em uma base de conhecimento AIML (3), levando em consideração que essa base é montada com o intuito de fornecer os instrumentos necessários para realizar uma conversação em linguagem natural, o mais próximo possível da empregada em uma conversa entre seres humanos.

Após a inserção e o processamento do texto terem sido concluídos, o usuário é direcionado para a interface de conversação com o ChatterEdu. O usuário pode iniciar a

conversação com uma mensagem de saudação ou diretamente com algo relacionado ao texto inserido (4). A mensagem é enviada para o *chatterbot* (5), que utiliza o interpretador Program AB para realizar a busca da mensagem nas bases AIML previamente criadas (6). Caso a entrada do usuário seja encontrada nas bases AIML, será retornada a resposta correspondente, caso contrário, será informado que não tem uma resposta para a pergunta inserida (7). O ChatterEdu então devolve a resposta obtida (8), fazendo o tratamento necessário para manter o histórico da conversa, que é mostrado para o usuário (9).

3.2.2 Análise morfológica do site Palavras

A base de toda a geração automática de perguntas da aplicação encontra-se nos papéis semânticos, obtidos através do site Palavras (BICK et al., 2016). Assim que o usuário informa o texto que deseja usar de base para a conversação, este é subdividido em frases. Feito isso, é criada uma requisição HyperText Transfer Protocol (HTTP) para cada frase do texto de entrada, que é enviada para o *parser* Palavras fazer o processamento utilizando a opção *semantic roles*.

O retorno da requisição traz a análise de cada palavra separadamente em um HyperText Markup Language (HTML). É feito um novo processamento a fim de dividir a sentença em um conjunto de palavras, cada uma com as seguintes informações: a palavra propriamente dita, a palavra base, os classificadores semânticos, a classe gramatical, a classe sintática, o papel semântico e se é um verbo na voz passiva ou não. Para exemplificar esse processamento, utiliza-se a frase “A Ásia é o continente mais populoso” e toma-se, como exemplo, o retorno do *parser* para a palavra “é” da Figura 8.

Figura 8 – Análise morfológica da frase "A Ásia é o continente mais populoso"

Flat structure

Enter Portuguese text to parse:
 A Ásia é o continente mais populoso

Parser: Semantic roles Visualization: Default

<β>
 a [o] **DET** F S @>N #1->2
 ásia [Ásia] **PROP** F S @SUBJ> §TH #2->3
 é [ser] <fmc> <vK> **V** PR 3S IND VFIN @FS-STA §PRED #3->0
 o [o] **DET** M S @>N #4->5
 continente [continente] <Ltop> **N** M S @<SC §ATR #5->3
 mais [mais] <quant> <KOMP> **ADV** @>A #6->7
 populoso [populoso] **ADJ** M S @N< #7->5 </β>

Fonte: Bick et al. (2016).

A partir do retorno obtido com o processamento *semantic roles*, tem-se as seguintes informações para a palavra “é”: (a) palavra: *é*; (b) palavra base: *ser*; (c) classificadores semânticos: *<fmc>* (verbo principal), *<vK>* (une o verbo com o sujeito predicativo); (d) classe gramatical: *v* (verbo); (e) complementos da classe gramatical: *PR* (tempo presente), *3S* (terceira pessoa do singular), *IND* (modo indicativo), *vFIN* (verbo no infinitivo); (f) classe sintática: *@FS-STA* (verbo principal); (g) papel semântico: *\$SPRED* (verbo principal); (h) voz passiva: falso.

Após a sentença estar devidamente dividida em palavras com suas respectivas análises morfológicas, a aplicação faz uma nova iteração sobre essa lista de palavras a fim de verificar qual o papel semântico de cada uma, para então definir o tipo de pergunta que será feita. Dentre os papéis semânticos existentes no *parser* Palavras, foram selecionados os que mais apareciam em textos da área de conhecimento em Geografia para serem tratados na implementação. A lista abaixo apresenta quais papéis temáticos foram considerados no desenvolvimento do presente projeto:

- a) AG: agente que executa uma ação, como por exemplo: Os barrigas-verdes (AG) moram em Santa Catarina;
- b) LOC: local ou localização, mostrado em: Santa Catarina fica na região sul (LOC);
- c) LOC-TMP: localização temporal, tal como: Blumenau sofreu uma grande enchente em 2008 (LOC-TMP);
- d) ORI-TMP: origem temporal, como no exemplo: Desde 1852 (ORI-TMP) ocorreram 64 enchentes em Blumenau;
- e) EXT: extensão ou quantidade, como na frase: O estado mede 95703 quilômetros (EXT) quadrado;
- f) EXT-TMP: extensão de uma medida temporal, exemplificado por: A tragédia durou por duas semanas (EXT-TMP);
- g) TH: tema principal da oração, como em: Blumenau (TH) é a terceira maior cidade de Santa Catarina.

Para um melhor entendimento de como é definida uma pergunta a partir do papel semântico, considera-se como exemplo o uso do papel semântico AG em uma frase. Logo, a pergunta criada deverá utilizar o pronome interrogativo “quem”. Já para uma palavra classificada como LOC, a pergunta correta a ser feita é “onde”. A lista completa com as palavras interrogativas (pronomes ou advérbios) utilizadas para cada papel semântico pode ser vista no Quadro 5.

Quadro 5 – Papéis semânticos e suas respectivas interrogativas

Papel semântico	Interrogativa	Frase de entrada	Pergunta formulada
AG – voz ativa	Quem	Os barrigas-verdes moram em Santa Catarina.	Quem mora em Santa Catarina?
AG – voz passiva	Por quem	Santa Catarina é habitada por barrigas-verdes.	Santa Catarina é habitada por quem?
LOC	Onde	Santa Catarina fica na região sul.	Onde Santa Catarina fica?
LOC-TMP	Quando	Blumenau sofreu uma grande enchente em 2008.	Quando Blumenau sofreu uma grande enchente?
ORI-TMP	Desde quando	Desde 1852 foram registradas 64 enchentes em Blumenau.	Desde quando foram registradas 64 enchentes em Blumenau?
EXT	Quanto	O estado mede 95703 quilômetros quadrado.	O estado mede quanto?
EXT-TMP	Quanto tempo	A tragédia durou por duas semanas.	A tragédia durou quanto tempo?
TH	O que	Florianópolis tem cerca de 421 mil habitantes.	O que tem cerca de 421 mil habitantes.?
TH – verbo “ser”	Qual	Blumenau é a terceira maior cidade de Santa Catarina.	Qual é a terceira maior cidade de Santa Catarina?
TH – humano	Quem	Ele foi empregado em diversas missões.	Quem foi empregado em diversas missões?

Fonte: elaborado pela autora.

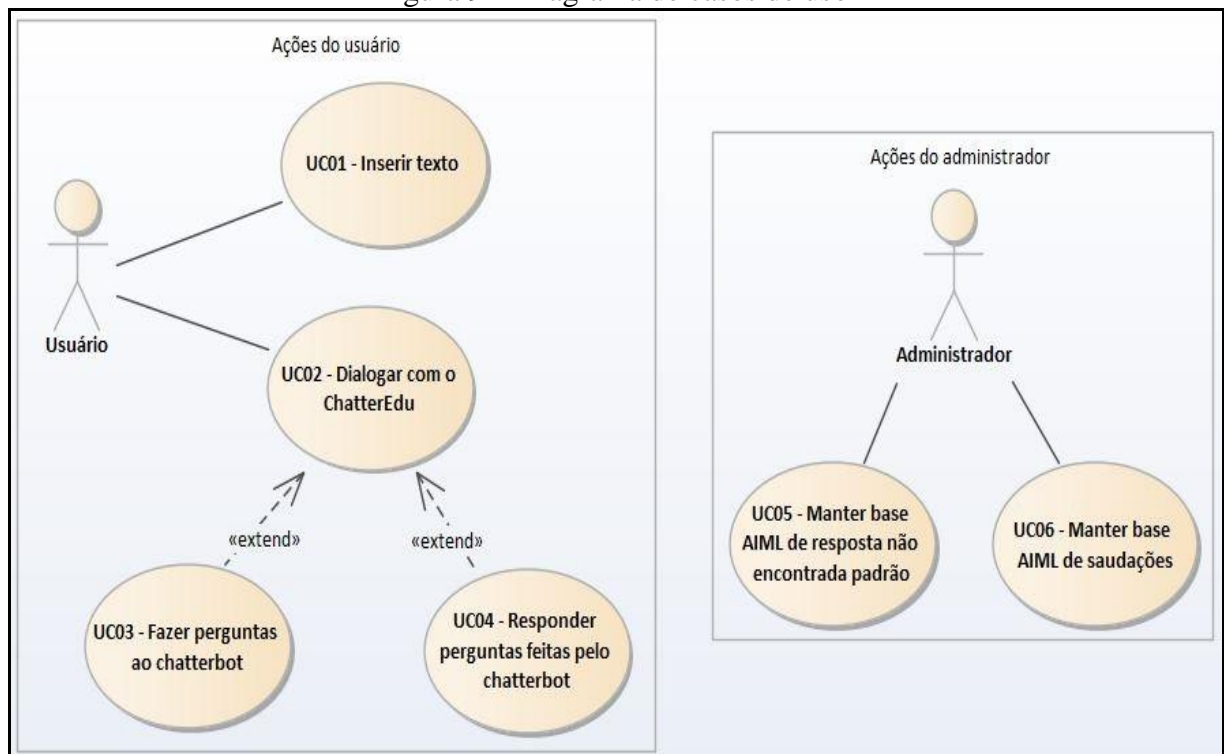
No Quadro 5 é possível notar que alguns papéis semânticos podem adotar mais de uma interrogativa dependendo de outros termos linguísticos presentes no texto de entrada. No caso do papel semântico AG, é verificado se a frase está na voz ativa ou passiva, para então definir se será utilizada a interrogativa “quem” ou “por quem”. Já o papel temático TH pode utilizar três palavras interrogativas diferentes em suas perguntas. Primeiramente, é verificado se a frase contém o verbo “ser”. Caso contenha, utiliza-se da regra geral da língua portuguesa e adota-se a interrogativa “qual”. Caso contrário, é feita uma nova requisição para o *parser* Palavras para realizar uma análise do tipo *full morphosyntactic parser* a fim de obter uma análise morfológica mais completa e verificar se existe algum humano na frase. Em caso positivo, então será utilizado o pronome interrogativo “quem” para fazer a pergunta. Caso nenhuma das situações supracitadas ocorra, será utilizada a pergunta padrão “o que”.

3.2.3 Casos de uso

O presente projeto contém seis casos de uso (em inglês Use Case - UC) que representam as principais funcionalidades do protótipo desenvolvido. O diagrama da Figura 9 foi feito utilizando a Unified Modeling Language (UML). Além disso, os UCs mais relevantes estão detalhados no Apêndice A.

O caso de uso *Inserir texto* representa a primeira ação do usuário, onde será informado o texto de Geografia que se deseja processar. Neste caso de uso é feito todo processamento linguístico necessário para transformar o texto de entrada em uma base de conhecimento AIML baseada em perguntas e respostas sobre o mesmo. Além da base gerada automaticamente, o ator Administrador mantém uma base de conhecimento com saudações (UC06) e outra com respostas padrão para quando o ChatterEdu não conseguir encontrar uma resposta para a pergunta formulada (UC05). Já o caso de uso *Dialogar com o ChatterEdu* permite que o usuário faça perguntas ao robô (UC03) ou responda perguntas sobre o texto inserido (UC04).

Figura 9 – Diagrama de casos de uso

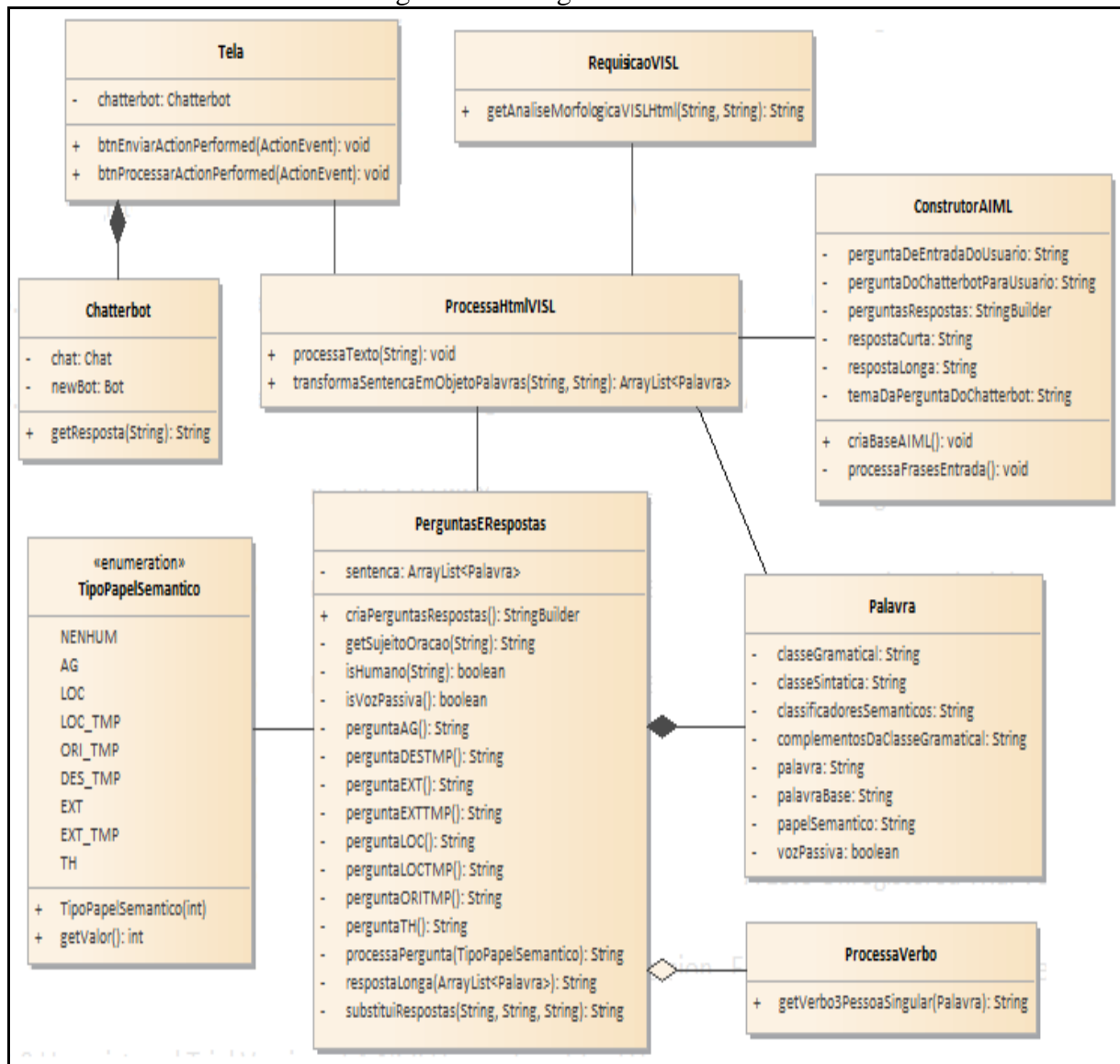


Fonte: elaborado pela autora.

3.2.4 Diagramas de classes

Na especificação do *chatterbot* desenvolveu-se o diagrama de classes apresentado na Figura 10. A fim de garantir uma melhor visualização, optou-se por omitir construtores, *gets*, *sets* e demais métodos secundários.

Figura 10 – Diagrama de classes



Fonte: elaborado pela autora.

A tela principal da aplicação exibe duas guias, a primeira para inserção do texto a ser processado a partir do método `btnProcessarActionPerformed`, já a segunda para conversação com o ChatterEdu utilizando o método `btnEnviarActionPerformed` para enviar a mensagem do usuário. A classe chamada `Chatterbot` instancia e utiliza os métodos encontrados na biblioteca do Program AB para fazer a interpretação das bases AIML. A classe `ProcessaHtmlVISL` é responsável por fazer o primeiro processamento do texto inserido e transformá-lo em uma lista de frases. Também realiza os tratamentos necessários para transformar o HTML retornado pela requisição feita ao *parser* Palavras em objetos da classe `Palavra`. A classe `RequisicaoVISL` faz a requisição ao site Palavras para obter a análise morfológica e o papel semântico das palavras.

Por sua vez, a classe `PerguntasERespostas` faz o processamento das palavras, objetos da classe `Palavra` armazenados no atributo `sentenca`, o qual representa uma frase morfológicamente analisada. Esta classe verifica quais papéis semânticos estão presentes na frase inserida e quais perguntas serão feitas a partir destes. Além disso, também cria perguntas, que podem ser feitas pelo usuário ou que o *chatterbot* poderá fazer ao usuário, para inserir na base de conhecimento AIML. Faz uso da classe `ProcessaVerbo` para efetuar os tratamentos necessário referentes à conjugação verbal em casos específicos.

Por fim, a classe `ConstrutorAIML` insere as perguntas e respostas retornadas em uma base de conhecimento AIML. Ademais, vale ressaltar que cada vez que ocorrer esse processamento, a base AIML criada será sobrescrita pelas novas informações adicionadas. Então, se o usuário inserir um texto e logo após inserir outro, o *ChatterEdu* conseguirá manter uma conversação apenas sobre o último texto inserido.

3.3 IMPLEMENTAÇÃO

Nesta seção são mostradas as ferramentas e técnicas utilizadas para implementar o presente projeto. Também é explanado sobre como é feito o processamento do texto de entrada e como utilizar a aplicação desenvolvida.

3.3.1 Técnicas e ferramentas utilizadas

Para implementar o trabalho proposto utilizou-se a linguagem de programação Java, dentro do ambiente de desenvolvimento Netbeans. Também foi necessário utilizar as bibliotecas `HttpCore` e `HttpClient` do Apache, para enviar e receber a requisição feita ao site Palavras. Além disso, o sistema foi desenvolvido para criar automaticamente e salvar em arquivos a base de conhecimento AIML com perguntas e resposta. Para poder ser realizada a conversação entre o *chatterbot* e o usuário, é necessário utilizar o interpretador Program AB para interpretar a base AIML. Este interpretador também é utilizado como uma biblioteca dentro do projeto no Netbeans.

3.3.2 Processamento do texto de entrada

A complexidade da aplicação desenvolvida está no processamento feito a partir do texto de entrada para transformá-lo em perguntas e respostas e, em seguida, criar a base de conhecimento AIML. O método que faz este processamento é o `processaTexto`, da classe `ProcessaHtmlVISL`. Primeiramente, o texto de entrada é dividido em sentenças, utilizando o ponto final para separá-las. Depois, o método `transformaSentencaEmObjetoPalavra`

divide cada sentença do texto em palavras (objetos da classe `Palavra`) com seus respectivos atributos (`palavra`, `palavraBase`, `classificadoresSemanticos`, `classeGramatical`, `complementosDaClasseGramatical`, `classeSintatica`, `papelSemantico`, `vozPassiva`), conforme exemplificado na seção 3.2.2. Para tanto, esse método chama o método `getAnaliseMorfologicaVISLHtml`, da classe `RequisicaoVISL`, para fazer uma requisição ao site Palavras tendo como parâmetros a sentença a ser processada e a constante "roles", informando que o processamento a ser realizado é o *semantic roles*. Além disso, as perguntas e respostas geradas pelo método `criaPerguntasRespostas`, da classe `PerguntasERespostas`, são adicionadas à variável `builderPerguntas`, um objeto da classe `StringBuilder`. O método `criaPerguntasRespostas` é responsável por verificar quais papéis semânticos estão presentes em cada sentença processada e quais perguntas serão feitas, assim como criar perguntas de diferentes modos para poder inserir na base de conhecimento AIML, como perguntas que podem ser feitas pelo usuário ou perguntas que o *chatbot* poderá fazer para o usuário. Ao final do processamento de todas as sentenças do texto, é instanciado um objeto da classe `ConstrutorAIML` para criar a base de conhecimento AIML a partir das perguntas e respostas geradas.

O método `getAnaliseMorfologicaVISLHtml`, da classe `RequisicaoVISL`, é o que de fato faz a requisição ao *parser* Palavras para obter a análise morfológica de uma sentença passada como parâmetro. Inicialmente, cria a URL que será utilizada para fazer a requisição, a qual é composta por: endereço do site Palavras, sentença a ser processada já formatada nos padrões HTML, tipo de processamento que será usado (*full morphosyntactic parse* ou *semantic roles*). Em seguida, faz a requisição e processa a resposta, que é uma lista de palavras morfológicamente analisadas e em formato HTML. Desta forma, a aplicação sempre necessita de acesso à internet para poder se conectar ao site Palavras.

Feitos os tratamentos necessários para obter a análise morfológica do texto de entrada, o processamento para transformar as sentenças em perguntas e respostas inicia-se na classe `PerguntasERespostas`. O método `criaPerguntasRespostas` usa o método `processaPergunta` para verificar o papel semântico de cada palavra e definir quais perguntas serão feitas. Cada papel semântico suportado pela aplicação desenvolvida tem seus tratamentos específicos e são processados por métodos diferentes. Para uma maior compreensão de como são montadas as perguntas e respostas a partir de um papel semântico, utiliza-se o processamento de LOC, através do método ilustrado no Quadro 6, que é o mais simples dentre os papéis temáticos tratados neste projeto.

Quadro 6 – Método perguntaLOC da classe PerguntasERespostas

```

1 private String perguntaLOC() {
2   String perguntaDeEntradaDoUsuario = "onde ^ ";
3   String temaDaPerguntaDoChatterbot = "^ sobre ^ onde ";
4   String perguntaDoChatterbotParaUsuario = "onde";
5
6   for (int i = 0; i < getSentenca().size(); i++) {
7     perguntaDoChatterbotParaUsuario += ' ' +
8                                     getSentenca().get(i).getPalavra();
9     if (!getSentenca().get(i).getPapelSemantico().isEmpty() &&
10        !getSentenca().get(i).getPapelSemantico().equalsIgnoreCase("LOC"))
11     { perguntaDeEntradaDoUsuario +=
12       getSentenca().get(i).getPalavra().trim() + ' ';
13       temaDaPerguntaDoChatterbot +=
14         getSentenca().get(i).getPalavra().trim() + ' ';
15     } else {
16       perguntaDeEntradaDoUsuario += " ^ ";
17       temaDaPerguntaDoChatterbot += " ^ ";
18     }
19   }
20   String respostaCurta = getSujeitoOracao("LOC").trim();
21
22   return substituiRespostas(perguntaDeEntradaDoUsuario,
23                             temaDaPerguntaDoChatterbot, respostaCurta)+
24     perguntaDoChatterbotParaUsuario.replace(respostaCurta, "") + "?";
25 }

```

Fonte: elaborado pela autora.

No caso do processamento do papel temático do tipo LOC, a palavra interrogativa que deve ser usada para fazer as perguntas é sempre “onde”. Além disso, todos os métodos que fazem o processamento de algum papel semântico possuem três variáveis a fim de armazenar a pergunta de entrada do usuário (linha 2), o tema da pergunta que o *chatterbot* deve fazer ao usuário (linha 3) e a pergunta do *chatterbot* para o usuário (linha 4).

A pergunta de entrada do usuário é utilizada quando o usuário quer perguntar algo sobre o texto para o ChatterEdu. Desta forma, para ser possível prever com maior precisão uma possível pergunta, é utilizado o sinal “^” (linha 2), que em uma base AIML representa um ponto da sentença onde pode haver nenhuma ou várias palavras quaisquer, permitindo uma maior flexibilidade nas perguntas reconhecidas. Por exemplo, se estiver gravada em uma base AIML a frase “Gostaria de saber onde fica Santa Catarina”, o usuário tem que entrar exatamente com essa frase para ser possível encontrar a resposta da sua pergunta. Já se a sentença gravada for “^ onde fica Santa Catarina ^” e o usuário entrar com uma variação da pergunta como “Eu gostaria de saber onde fica Santa Catarina para estudar para prova” ou “Quero saber onde fica Santa Catarina”, ambas são reconhecidas.

O tema da pergunta do *chatterbot* é utilizado quando o usuário quer que o robô faça perguntas para ele. Sendo assim, é criada uma frase utilizando a mesma ideia descrita para a pergunta de entrada do usuário, mas agora utilizando a palavra “sobre” na frente. O exemplo

anterior ficaria “^ sobre ^ onde fica Santa Catarina ^”. Então, sempre que o usuário quiser responder a perguntas, deve entrar com uma frase tal como: “Quero falar sobre onde fica Santa Catarina”. Nesse caso, o retorno que o ChatterEdu fornece é uma pergunta referente ao assunto solicitado pelo usuário.

Já a pergunta do *chatterbot* para o usuário é a mais simples de ser elaborada. Isto porque não há necessidade de substituir nenhuma palavra por coringas da linguagem AIML, uma vez que a pergunta que o ChatterEdu deve fazer para o usuário deve estar completa. Desta forma, é adicionada a interrogativa “onde” no início da frase e em seguida é concatenada toda a frase afirmativa que foi recebida por parâmetro.

A partir das considerações anteriores, pode-se definir como deve ser feita a elaboração das perguntas. Para determinar quais palavras podem ser substituídas pelo acento circunflexo, é verificado se a palavra tem um papel semântico (Quadro 6 – linhas 9 e 10). Se tiver, assume-se que é uma palavra de importância para a contextualização da pergunta e, portanto, deve ser mantida (linhas 11 a 14). Caso contrário, a palavra é substituída pelo coringa da linguagem AIML (linhas 16 e 17). Entretanto, fazendo o processamento das perguntas simplesmente como descrito, a resposta seria inserida junto da pergunta, já que a resposta tem um papel semântico definido, que é o LOC. Por exemplo, na frase de entrada “Santa Catarina fica no sul (LOC)” a pergunta mais simples ficaria “onde Santa Catarina fica no sul”. Para evitar isto, são utilizados os métodos `getSujeitoOracao` e `substituiRespostas` (linhas 20, 22 e 23). O primeiro método retorna a resposta curta extraída da frase de entrada que, neste caso, seria “no sul”. Já o segundo método é utilizado para retirar a resposta curta das frases que foram criadas contendo os coringas da linguagem AIML. E, por fim, é utilizado o método `replace` para substituir a resposta curta por uma *string* vazia na pergunta que o *chatterbot* fará para o usuário. Os três tipos de perguntas são concatenadas e retornadas pelo método.

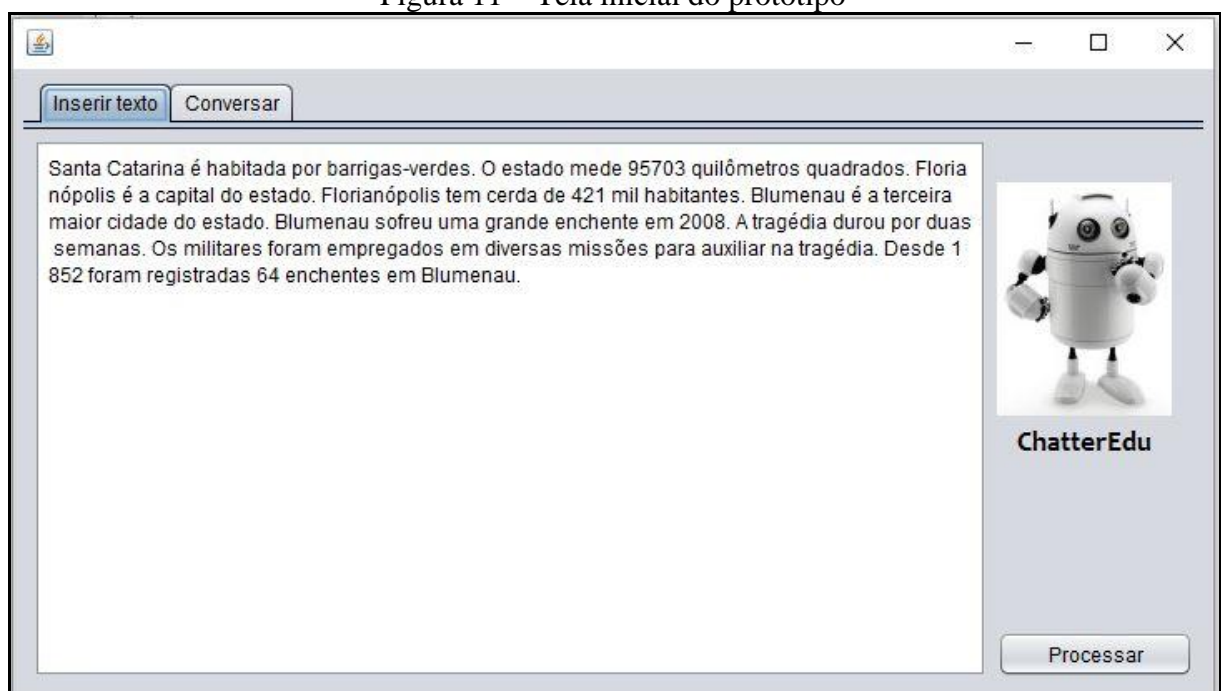
O último passo realizado pelo processamento do texto de entrada é criar a base AIML. Para isto, é utilizada a classe `ConstrutorAIML` que recebe as perguntas e respostas criadas e as transforma em uma base de conhecimento

3.3.3 Operacionalidade da implementação

O ChatterEdu foi desenvolvido como uma aplicação desktop e necessita ser instalado no computador para poder utilizá-lo. Também é necessária conexão com a internet, pois durante seu processamento é feita uma requisição ao site Palavras para obter a análise morfológica do texto de entrada. A tela principal do aplicativo, mostrada na Figura 11, é onde

o usuário insere o texto de entrada. Ao clicar no botão *Processar*, serão feitos os processamentos necessários para transformar o texto de entrada em perguntas e respostas, para então gerar a base de conhecimento AIML (vide Apêndice B). Ao final, é retornada uma mensagem avisando se o processamento foi ou não realizado com sucesso. Em seguida, o usuário é redirecionado para a aba *Conversar*, onde pode iniciar uma conversa com o *chatterbot*. É importante salientar que o texto de entrada deve ser um texto com linguagem simples, sendo que cada oração deve ser composta por sujeito, verbo e predicado, para viabilizar a extração das perguntas e respostas.

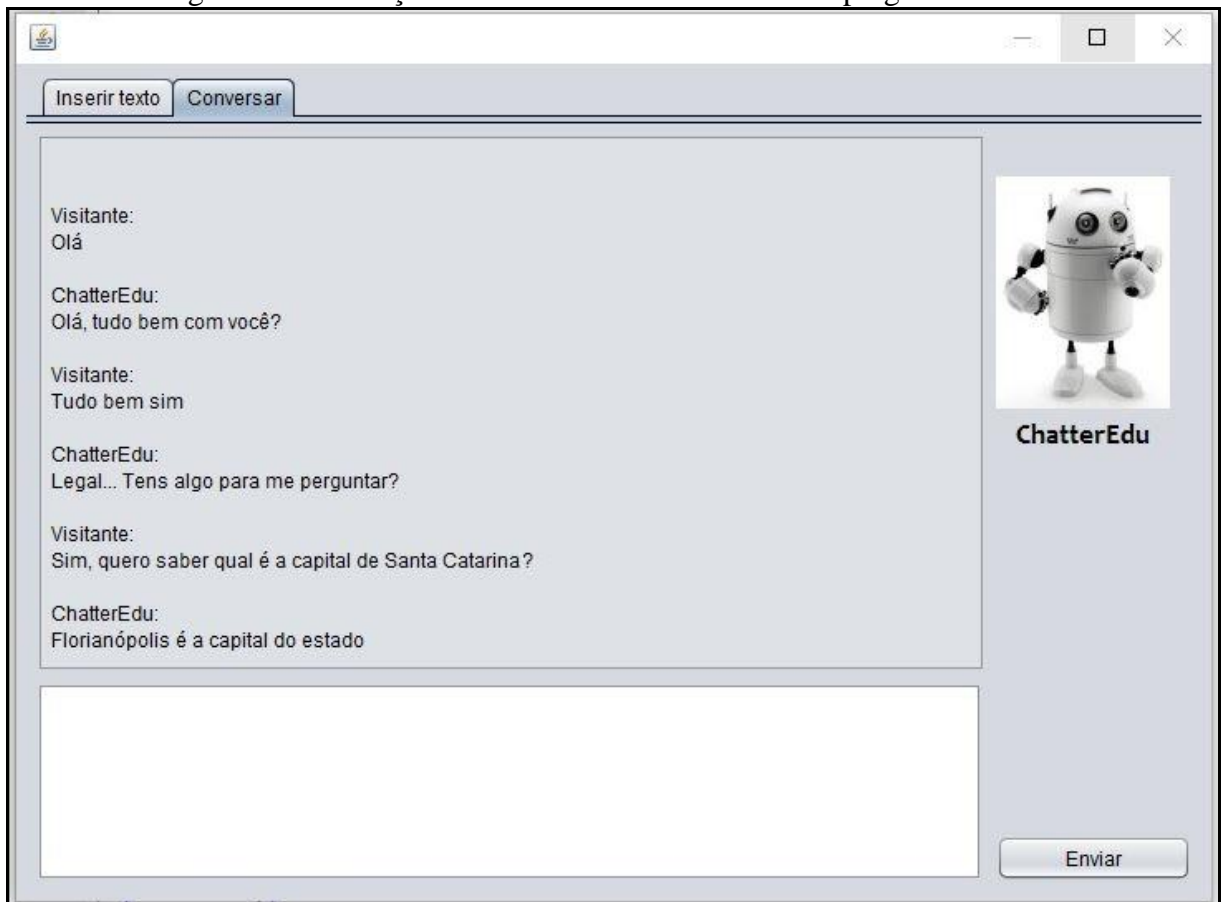
Figura 11 – Tela inicial do protótipo



Fonte: elaborado pela autora.

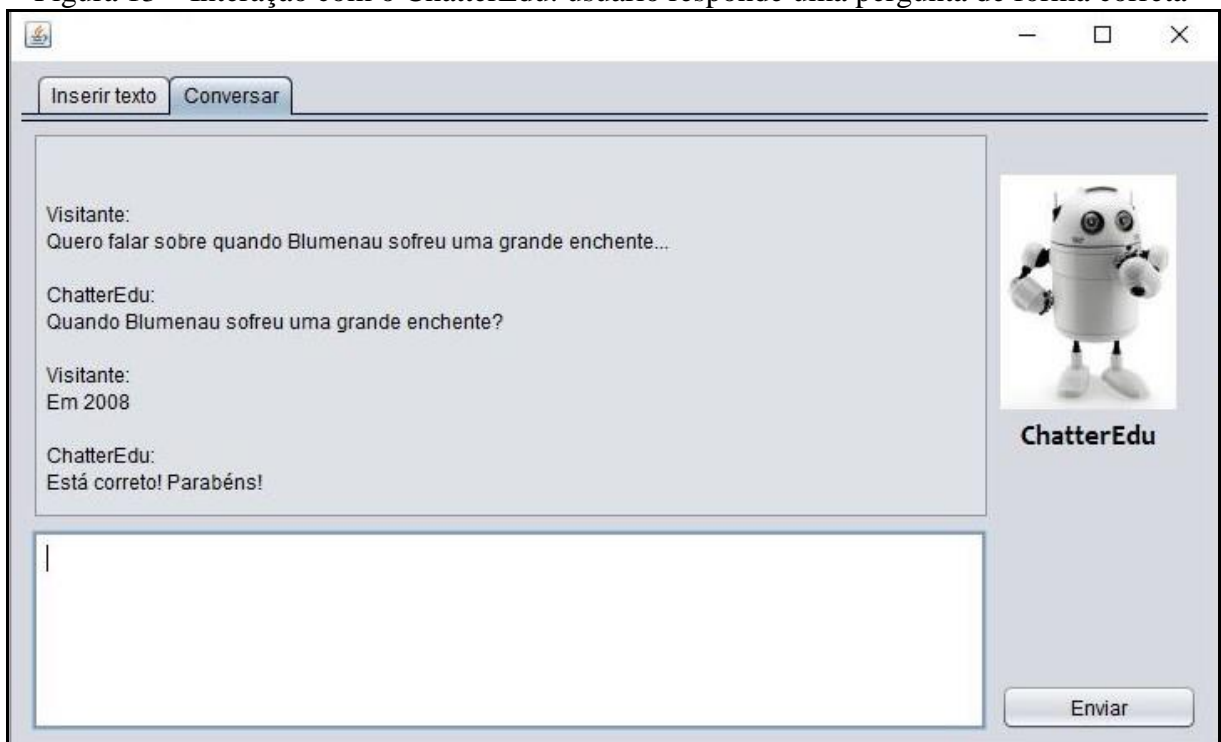
Após o processamento ter sido realizado, pode ser iniciada a conversação com o *chatterbot*. O usuário poderá utilizar saudações durante o início e final da conversa. Além disso, poderão ser formuladas perguntas para o robô a respeito do texto previamente inserido, ou o usuário poderá expressar sua vontade de falar sobre algum tema específico do texto, sendo que, nesse caso, o *chatterbot* fará as perguntas e verificará se a resposta informada está correta ou não. Também é possível que o usuário peça ao ChatterEdu que faça alguma pergunta aleatória sobre o texto de entrada, sendo também verificado se a resposta informada está correta ou não. As figuras seguintes ilustram exemplos de possíveis interações com o robô. Na Figura 12 o usuário faz a pergunta sobre o texto, obtendo uma resposta do ChatterEdu. Já na Figura 12 e na Figura 14, o usuário solicita ao *chatterbot* que este lhe faça perguntas. Na primeira situação, para a pergunta feita pelo robô, o usuário dá uma resposta correta, enquanto na segunda, fornece uma resposta errada.

Figura 12 – Interação com o ChatterEdu: usuário faz pergunta ao robô



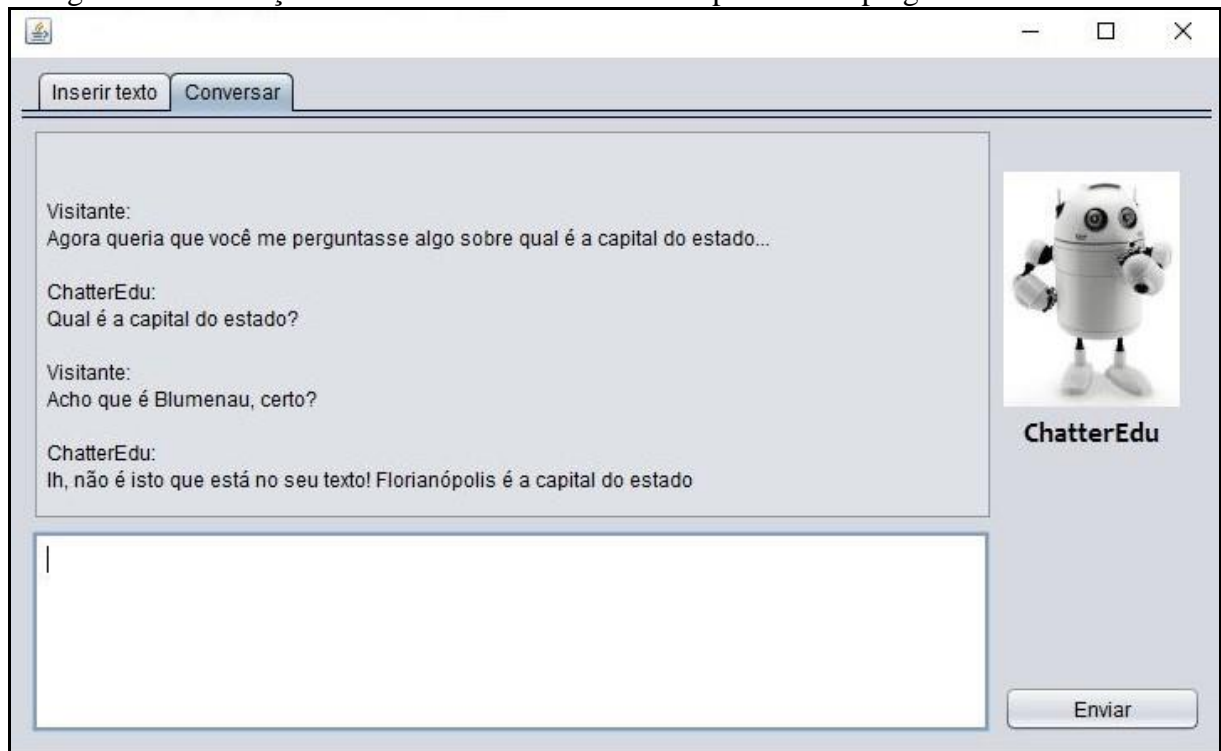
Fonte: elaborado pela autora.

Figura 13 – Interação com o ChatterEdu: usuário responde uma pergunta de forma correta



Fonte: elaborado pela autora.

Figura 14 – Interação com o ChatterEdu: usuário responde uma pergunta de forma errada



Fonte: elaborado pela autora.

3.4 ANÁLISE DOS RESULTADOS

O presente trabalho atingiu, de uma maneira geral, os objetivos propostos. O único objetivo que teve que ser modificado foi a interface web. Desde o início foi feita a interface totalmente voltada para web e já estava praticamente pronta, entretanto todas as adaptações necessárias eram muito custosas por conta da falta de experiência da autora com tecnologias web. Desta forma, optou-se por modificar a interface e fazê-la voltada para aplicações desktop, que ficou funcional e o usuário consegue fazer toda a interação com o ChatterEdu em língua portuguesa. Também é possível inserir textos da educação básica da área de conhecimento de Geografia e, a partir deste, são geradas perguntas e respostas que são tratadas e adicionadas à base de conhecimento AIML. Desta forma, possibilita-se que o usuário mantenha uma conversação com o *chatterbot* sobre o texto inserido, sendo possível fazer perguntas ao robô ou responder às perguntas feitas por ele. No entanto, um texto para ser processado deve estar gramaticalmente correto, sendo que todas as sentenças devem ser compostas por sujeito, verbo e predicado. A partir dos papéis semânticos das palavras que compõem as sentenças, determinam-se quais pronomes ou advérbios interrogativos serão usados para gerar as perguntas. O ChatterEdu usa os seguintes: “o que”, “qual”, “quem”, “por quem”, “onde”, “quando”, “desde quando” e “quanto tempo”. Apesar da limitação quanto às perguntas geradas e aos papéis semânticos tratados, o uso desses permitiu que o trabalho

desenvolvido processe e extraia perguntas e respostas de textos de qualquer área de conhecimento, não se restringindo a conhecimentos de Geografia, conforme havia sido proposto.

Entretanto, mesmo com as restrições estabelecidas quanto aos papéis semânticos que o *chatbot* trata e, conseqüentemente, o pronome ou advérbio interrogativo usado para formar as perguntas, existem sentenças que não são processadas adequadamente, gerando perguntas com erros gramaticais, que limitam o uso do ChatterEdu. Cita-se como exemplo a formulação de perguntas com “o que”, pois quando este termo aparece no final da sentença interrogativa deve ser acentuado. Porém, o tratamento feito pelo protótipo não faz diferenciação em relação ao posicionamento do termo “o que” na pergunta. Outros erros gramaticais que podem ocorrer são em relação às conjugações verbais. Em alguns casos, as palavras interrogativas são invariáveis, portanto mesmo que no texto de entrada a frase utilizada para gerar a pergunta esteja em um determinado tempo verbal e com determinadas flexões, o verbo utilizado na pergunta não poderá sofrer as mesmas variações que tem na frase afirmativa. Para contornar este problema foi feito um tratamento para quando o verbo estiver no plural e nos tempos verbais presente, pretérito perfeito, pretérito imperfeito e pretérito-mais-que-perfeito. Com isso, todos os verbos regulares são tratados, assim como alguns verbos irregulares (como os verbos ser, ir, estar e dar). Todavia, os demais verbos irregulares ficaram sem o devido tratamento, bem como os outros tempos verbais.

Outro desafio encontrado diz respeito à elaboração das perguntas e respostas a partir dos papéis semânticos. Por exemplo, na frase “O clima no Nordeste é quente”, deveria ser feita a pergunta “Onde o clima é quente?”, tendo como resposta “No Nordeste”. Entretanto, conforme o processamento implementado, o *chatbot* traz a pergunta incompleta “Onde é quente?” e a resposta “O clima no Nordeste”. O contrário desta situação também pode ocorrer, isto é, o robô pode retirar um trecho menor para a resposta e formular uma pergunta que ainda tenha parte da resposta embutida. Ainda quanto aos papéis temáticos, também foi percebida uma certa dificuldade em definir a interrogativa correta para ser utilizada em frases com o papel semântico TH. Como TH diz respeito ao tema, pode variar muito de uma oração para outra. Sendo assim, optou-se por utilizar regras gerais da gramática que se aplicam para a maioria das palavras, mas podem haver exceções. Desta forma, foram tratados os casos em que a interrogativa varia para “quem” ou “qual” e nos demais casos utiliza-se o padrão “o que”, que é o mais assertivo na maioria dos casos testados. As diferenças encontradas em frases que tem o mesmo papel semântico TH foram apresentadas no Quadro 5.

Por fim, observou-se demora em processar os textos de entrada, pois para efetuar a análise morfológica e determinar os papéis semânticos é feita uma requisição ao site Palavras para cada frase. Assim, por exemplo, para processar uma frase com 7 palavras, a aplicação leva em média 6 segundos, já um texto com 41 palavras, leva em torno de 15 segundos para fazer o processamento e redirecionar o usuário para a conversação. Ressalta-se que podem haver variações dependendo da intensidade do sinal de internet no momento da requisição. Além disso, a linguagem AIML tem suas próprias limitações que forçam o usuário a fazer as perguntas e respostas utilizando uma linguagem muito similar à usada no texto de entrada. Por conta disso, mesmo que o usuário fale sobre algo do texto inserido, em alguns momentos o *chatbot* não vai conseguir encontrar uma resposta à pergunta feita pois não está exatamente como a pergunta gravada na base AIML.

Em relação aos trabalhos correlatos, levantaram-se as principais características desejáveis ou presentes nos *chatbots* descritos no capítulo anterior e elaborou-se o quadro comparativo abaixo (Quadro 7).

Quadro 7 – Comparativo entre o ChatterEdu e os trabalhos correlatos

<i>chatbot</i>	ChatterEdu	Elektra	Tuxbot	ChatterDóris
características				
utiliza bases AIML	sim	sim	sim	Sim
gera as bases de conhecimento automaticamente	sim	não	sim	Não
utiliza buscas on-line para montar a base de conhecimento	não	não	sim	Não
utiliza <i>corpus</i> ou <i>corpora</i>	não	não	sim	Não
consegue conversar sobre assuntos diversos	não	não	não	Não
é voltado para a área educacional	sim	sim	sim	Sim

Fonte: elaborado pela autora.

É possível notar alguns pontos em comum em todos os trabalhos apresentados, como a utilização de bases AIML e todos serem voltados para a área educacional. Entretanto, verifica-se que, dentre os trabalhos supracitados, o Tuxbot e o trabalho desenvolvido são mais flexíveis por gerarem as bases de conhecimento, enquanto o Elektra e o ChatterDóris são mais limitados por conta de suas bases de conhecimento serem fixas. Porém, comparando o presente trabalho com o Tuxbot, verifica-se que o trabalho correlato usa mecanismos mais avançados para criar suas bases de conhecimento. O Tuxbot é o único dos *chatbots* que gera sua base de conhecimento a partir de textos pesquisados na internet, enquanto o ChatterEdu necessita que o usuário insira o texto com o assunto que deseja conversar e os

outros dois trabalhos não comportam nenhum tipo de inserção, ou seja, só conseguem conversar sobre temas que se encontram nas bases AIML criadas manualmente pelo administrador do *chatterbot*. Além disso, o Tuxbot é capaz de encontrar textos em buscadores on-line e transformá-los em *corpus* para então criar sua base de conhecimento. Observa-se que nenhum dos quatro *chatterbots* consegue conversar sobre qualquer tipo de assunto. Todos devem ter um assunto especificado para ser possível realizar a consulta nas bases AIML. O Tuxbot, por exemplo, tira dúvidas sobre Linux, o ChatterDóris conversa sobre assuntos abordados em sala de aula, já o Elektra responde sobre física, redes de computadores e internet. O ChatterEdu, por outro lado, é capaz de conversar sobre qualquer assunto inserido a partir do texto de entrada, embora inicialmente tenha sido projetado para processar textos da educação básica na área de conhecimento em Geografia.

4 CONCLUSÕES

Durante o desenvolvimento desse trabalho, criou-se uma aplicação desktop que interage com o usuário em língua portuguesa, possibilita o processamento de um texto da educação básica na área de Geografia, faz perguntas e traz respostas a partir do texto inserido e possibilita que o usuário faça ou responda perguntas sobre o texto numa conversação com o ChatterEdu. Portanto, a maioria dos objetivos elencados foram atingidos com sucesso e, apesar de ainda existir alguns pontos a serem melhorados, o resultado foi satisfatório. O único objetivo que não foi atingido foi o desenvolvimento de uma interface web, que certamente enriqueceria o trabalho pois permitiria que o ChatterEdu estivesse disponível on-line.

Entretanto, houveram alguns momentos difíceis para alcançar os objetivos propostos. O primeiro obstáculo enfrentado foi a escolha da ferramenta para fazer a análise morfológica. No princípio esperava-se utilizar uma ferramenta chamada Rembrandt para extração dos papéis semânticos. Porém, depois de pesquisas e tentativas, percebeu-se que não seria possível utilizá-la, pois as documentações eram antigas, a ferramenta não recebia atualizações há alguns anos e não foi possível instalá-la no computador utilizado para o desenvolvimento do *chatbot*. A partir de então, foram feitas pesquisas e testes até encontrar uma ferramenta que suprisse as necessidades do trabalho proposto. Então foi definido que seria utilizado o *parser* Palavras, que se tornou peça fundamental para a implementação do protótipo e atendeu os requisitos para o processamento linguístico realizado. Além disso, a geração de perguntas e respostas automatizadas para criar bases de conhecimento AIML na língua portuguesa ainda é uma tarefa pouco explorada. A complexidade da estrutura linguística do português também impactou no desenvolvido do ChatterEdu, de tal forma que existem erros ortográficos e de concordância que não foram tratados.

Por outro lado, a ferramenta mostrou-se mais abrangente do que foi inicialmente proposto. A princípio a ferramenta deveria processar textos da educação básica na área de conhecimentos de Geografia. Mas, fazendo testes com textos de outras áreas de conhecimento, obteve-se bons resultados, embora as sentenças a serem processadas não devem ser muito complexas em sua estrutura para não comprometer o desempenho do *chatbot*.

Ao final do desenvolvimento do projeto, acredita-se que este possa contribuir para a área de processamento de linguagem natural, visto que diversos trabalhos pesquisados que utilizam *chatbot* são feitos com bases de conhecimento fixas, justamente devido à complexidade linguística do idioma português. A partir das ferramentas e técnicas aplicadas

neste trabalho, conclui-se que é possível utilizá-lo como base para estudos relacionados com a geração de perguntas e respostas automáticas e criação de bases de conhecimento AIML dinâmicas.

Outrossim, presume-se que este trabalho também seja relevante para a área educacional, por ter sido desenvolvido visando principalmente o manuseio por alunos da educação básica. Com a tecnologia tornando-se cada vez mais comum no cotidiano das crianças, é necessário que os métodos de ensino também evoluam, renovem-se e acompanhem os avanços tecnológicos, a fim de manter os alunos das novas gerações interessados nos processos de ensino-aprendizagem. Ferramentas como a desenvolvida podem fazer parte desse processo.

4.1 EXTENSÕES

O projeto apresentado nesta monografia atingiu os objetivos propostos. Entretanto, existem pontos que podem ser melhorados e novas funcionalidades que podem ser implementadas. São eles:

- a) refatorar o código-fonte e buscar métodos de melhorar a performance da aplicação;
- b) tratar a geração da base AIML para não sobrescrevê-la a cada execução, possibilitando manter as perguntas e respostas já geradas;
- c) refinar o tratamento dos papéis semânticos já desenvolvidos neste trabalho, a fim de formular perguntas e respostas sem erros gramaticais;
- d) adicionar tratamento para outros papéis semânticos para aumentar as possibilidades de perguntas a serem feitas e respondidas pelo *chatbot*;
- e) incorporar um corretor gramatical no processamento da entrada do usuário, para possibilitar a criação das bases de conhecimento AIML mesmo quando o texto de entrada contenha erros gramaticais;
- f) utilizar as perguntas e respostas geradas para formular um questionário com questões objetivas, onde o usuário tenha uma pergunta e algumas opções de respostas com apenas uma das alternativas correta;
- g) migrar a aplicação desktop para uma página web ou um aplicativo para smartphones;
- h) desenvolver um mecanismo onde professores possam aplicar provas e medir os conhecimentos de seus alunos a partir da aplicação disponibilizada;
- i) desenvolver um controle para não repetir as perguntas já realizadas em uma

mesma conversa;

- j) desenvolver uma interface gráfica que permita fazer edições e ajustes nas perguntas e respostas sugeridas pelo *chatterbot* antes de iniciar a conversação;
- k) realizar testes da aplicação em sala de aula juntamente com alunos e professores para validar a ferramenta.

REFERÊNCIAS

- BARBOSA, Alexandre F. (Org.) **Pesquisa sobre o uso das tecnologias de informação e comunicação no Brasil: TIC educação 2011**. São Paulo: Comitê Gestor da Internet no Brasil, 2012. Disponível em: <<http://www.cetic.br/media/docs/publicacoes/2/tic-educacao-2011.pdf>>. Acesso em: 30 mar. 2016.
- BICK, Eckhard et al. **The parsing system: Palavras**. [S.l.], [2016]. Disponível em: <<http://beta.visl.sdu.dk/visl/pt/parsing/automatic/parse.php>>. Acesso em: 09 nov. 2016.
- COMARELLA, Rafaela L.; CAFÉ, Ligia M. A. Chatterbot: conceito, características, tipologia e construção. **Informação & Sociedade: estudos**, João Pessoa, v. 18, n. 2, p. 55-67, maio/ago. 2008. Disponível em: <<http://www.ies.ufpb.br/ojs/index.php/ies/article/view/1758>>. Acesso em: 30 mar. 2016.
- GALVÃO, Adjamir M. **Persona-AIML: uma arquitetura para desenvolver chatterbots com personalidade**. 2003. 139 f. Dissertação (Mestrado em Ciência da Computação) - Universidade Federal de Pernambuco, Recife, 2003. Disponível em: <<http://www.liber.ufpe.br/teses/arquivo/20040507160708.pdf>>. Acesso em: 01 abr. 2016.
- GUPTA, Poonam; GUPTA, Vishal. A survey of text question answering techniques. **International Journal of Computer Applications**, New York, v. 53, n. 4, p. 1-8, Sept. 2012. Disponível em: <<http://research.ijcaonline.org/volume53/number4/pxc3882030.pdf>>. Acesso em: 11 nov. 2016.
- KÜHLEIS, Richard. et al. ChatterDóris: um chatterbot que expressa emoções. In: **COMPUTER ON THE BEACH, 3.**, 2012, Florianópolis. **Anais...** [S.l.]: UNIVALI, 2012. p.1-10. Disponível em: <<http://siaiap32.univali.br/seer/index.php/acotb/article/view/6545/3691>>. Acesso em: 02 abr. 2016.
- LE, Nguyen-Thinh; KOJIRI, Tomoko; PINKWART, Niels. Automatic question generation for educational applications: the state of art. In: DO, Tien; THI, Hoai A.; NGUYEN, Ngoc T. (Eds.). **Advanced computational methods for knowledge engineering**. Berlim: Springer, 2014. p 325-338. Disponível em: <https://cses.informatik.hu-berlin.de/pubs/2014/iccsama/automatic_question_generation_for_educational_applications_the_state_of_art.pdf>. Acesso em: 11 nov. 2016.
- LEONHARDT, Michelle D. et al. **Elektra: um chatterbot para uso em ambiente educacional**. [Porto Alegre], 2003. Disponível em: <<http://penta3.ufrgs.br/~elektra/info/artigos/chatterbot-Elektra%5B1%5D.PDF>>. Acesso em: 02 abr. 2016.
- MOURA, Thiago J. M. **Um chatterbot para aquisição automática de perfil do usuário**. 2003. 127 f. Dissertação (Mestrado em Ciência da Computação) - Universidade Federal de Pernambuco, Recife, 2003. Disponível em: <<http://www.liber.ufpe.br/teses/arquivo/20050228150106.pdf>>. Acesso em: 01 abr. 2016.
- OLIVEIRA, Bruno. **Chatterbot para esclarecimento de dúvidas sobre as formas de ingresso em cursos da FURB**. 2015. 65 f. Trabalho de Conclusão de Curso (Bacharelado em Ciência da Computação) – Universidade Regional de Blumenau, Blumenau. Disponível em: <http://dsc.inf.furb.br/arquivos/tccs/monografias/2015_1_bruno-de-oliveira_monografia.pdf>. Acesso em: 01 abr. 2016.
- OLIVEIRA, Cláudio L. V. **Visão geral sobre Artificial Intelligence Markup Language (AIML)**. [S.l.], [2010?]. Disponível em: <http://www.profclaudio.com.br/arquivos/AIML-Visao_Geral.pdf>. Acesso em: 03 abr. 2016.

PILASTRI, André L.; BREGA, José R. F. **Chatterbot com interatividade ao avatar encapsulado no ambiente virtual second life usando a base de conhecimento em AIML.** [S.l.], 2009. Disponível em: <<http://www.lbd.dcc.ufmg.br/colecoes/wrva/2009/0027.pdf>>. Acesso em: 01 abr. 2016.

SCARTON, Carolina E. **VerbNet.Br:** construção semiautomática de um léxico verbal online e independente de domínio para o português do Brasil. 2013. 242 f. Dissertação (Mestrado em Ciências da Computação e Matemática Computacional) – Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, São Carlos. Disponível em: <<http://www.teses.usp.br/teses/disponiveis/55/55134/tde-19042013-160640/pt-br.php>>. Acesso em: 11 nov. 2016.

SOARES, Eduardo C.; MENUZZI, Sérgio de M. Introduzindo e problematizando papéis temáticos e hierarquias temáticas: uma questão de interfaces. **Signo**, Santa Cruz do Sul, v. 35, n. 59, p. 13-43, jul./dez. 2010. Disponível em: <<https://online.unisc.br/seer/index.php/signo/article/view/1432>>. Acesso em: 11 nov. 2016.

TEIXEIRA, Sérgio. **Chatterbots:** uma proposta para a construção de bases de conhecimento. 2005. 100 f. Dissertação (Mestrado em Informática) – Universidade Federal do Espírito Santo, Vitória. Disponível em: <<http://www.multicast.com.br/sergio/tuxbot-dissertacao-mestrado-sergio-teixeira.pdf>>. Acesso em: 02 abr. 2016.

APÊNDICE A – Detalhamento dos casos de uso

A seguir é apresentado o detalhamento dos casos de uso UC01: Inserir texto e do UC02: Dialogar com o ChatterEdu.

Quadro 8 – Caso de uso UC01: Inserir texto

UC01 – Inserir texto: permite que o usuário insira, a partir de uma interface gráfica, um texto sobre o qual quer conversar com o <i>chatterbot</i>	
Descrição	Permite inserir um texto sobre conhecimentos na área de Geografia
Ator	Usuário
Cenário principal	<ol style="list-style-type: none"> 1. O usuário digita o texto sobre o qual deseja conversar. 2. O protótipo faz o processamento do texto e o transforma em perguntas e respostas. 3. O protótipo cria a base AIML a partir do processamento realizado. 4. O protótipo redireciona o usuário para a conversação.
Pré-condições	O texto inserido não deve conter erros gramaticais e cada frase deve conter sujeito, verbo e predicado, nessa ordem.
Pós-condições	Base AIML gerada e carregada no protótipo.

Fonte: elaborado pela autora.

Quadro 9 – Caso de uso UC02: Dialogar com o ChatterEdu

UC02 – Dialogar com o ChatterEdu: permite que o usuário realize perguntas ao robô ou responda as perguntas elaboradas pelo ChatterEdu	
Descrição	Permite realizar ou responder perguntas ao ChatterEdu
Ator	Usuário
Cenário principal	<ol style="list-style-type: none"> 1. O usuário digita as saudações iniciais. 2. O usuário faz uma pergunta sobre o texto ao <i>chatterbot</i>. 3. ChatterEdu fornece uma resposta à pergunta formulada.
Fluxo alternativo 01	<ol style="list-style-type: none"> 1. O usuário descreve sobre qual tema gostaria de ser perguntado. 2. O robô faz a pergunta ao usuário. 3. O usuário responde à pergunta ao <i>chatterbot</i>. 4. ChatterEdu retorna a informação de que a pergunta foi ou não respondida corretamente.
Pré-condições	Ter um texto inserido a partir do UC01. A conversação não deve conter erros.
Pós-condições	Histórico da conversação apresentado no protótipo

Fonte: elaborado pela autora.

APÊNDICE B – Detalhamento das bases de conhecimento AIML

No Quadro 10 é apresentado um trecho da base de conhecimento AIML gerada a partir da frase “Blumenau sofreu uma grande enchente em 2008.” do texto de entrada exemplificado na Figura 11 (seção 3.3.3). Nas linhas 2 a 10 foi definido como o usuário faz uma pergunta ao ChatterEdu sobre o assunto em questão. Cita-se como exemplo “Quero que você me fale quem sofreu com enchente em 2008...” ou outras variações da frase, desde que contenha “^ quem ^ sofreu ^ enchente ^ 2008”, onde ^ pode ser qualquer sequência de palavras (zero ou mais). Nesse caso, o *chatterbot* pode dar como resposta “Blumenau” (linha 6) ou “Blumenau sofreu uma grande enchente em 2008” (linha 7). Já nas linhas 12 a 15 tem-se a categoria para quando o usuário solicita ao ChatterEdu que lhe faça uma pergunta sobre o assunto, tal como “Gostaria de falar sobre quem sofreu muito com uma enchente de proporções enormes no ano de 2008”. Nesse caso a pergunta será “Quem sofreu uma grande enchente em 2008?” (linha 14). Nas linhas 17 a 32 e nas linhas 34 a 51, define-se a sequência do diálogo, ou seja, para a pergunta anterior, será verificado se a resposta informada pelo usuário foi “Blumenau” (linha 18). Em caso positivo, o ChatterEdu fornece uma resposta aleatória (linhas 21 a 30) parabenizando-o. Em caso negativo, ou seja, para qualquer outra resposta (linha 35), o *chatterbot* informa também de forma aleatória que a resposta não está correta (linhas 38 a 49). A mesma lógica é usada para definir a pergunta usando o pronome interrogativo quando (linhas 53 a 102).

Quadro 10 – Base de conhecimento gerada

1	...
2	<category>
3	<pattern>^ quem ^ sofreu ^ enchente ^ 2008</pattern>
4	<template>
5	<random>
6	Blumenau
7	Blumenau sofreu uma grande enchente em 2008
8	</random>
9	</template>
10	</category>
11	
12	<category>
13	<pattern>^ sobre ^ quem ^ sofreu ^ enchente ^ 2008</pattern>
14	<template>Quem sofreu uma grande enchente em 2008?</template>
15	</category>
16	
17	<category>
18	<pattern>^ Blumenau ^</pattern>
19	<that>Quem sofreu uma grande enchente em 2008</that>
20	<template>
21	<random>
22	Isso mesmo! Você está indo muito bem nos estudos!
23	Parabéns, é isso mesmo!
24	Está correto! Parabéns!
25	Acertou em cheio hein!

```

26 <li>Opa, acertou! Você é fera!</li>
27 <li>Está certo! Desse jeito você vai tirar dez na prova deste
28 assunto!</li>
29 <li>Certíssimo, meus parabéns!</li>
30 </random>
31 </template>
32 </category>
33
34 <category>
35 <pattern>^</pattern>
36 <that>Quem sofreu uma grande enchente em 2008</that>
37 <template>
38 <random>
39 <li>Ish, na realidade não é bem isso... Blumenau sofreu uma grande
40 enchente em 2008</li>
41 <li>Ah, não foi dessa vez! Blumenau sofreu uma grande enchente em
42 2008</li>
43 <li>Puxa, não é essa a resposta correta... Blumenau sofreu uma grande
44 enchente em 2008</li>
45 <li>Ops, dessa vez você não acertou. Blumenau sofreu uma grande
46 enchente em 2008</li>
47 <li>Ih, não é isto que está no seu texto! Blumenau sofreu uma grande
48 enchente em 2008</li>
49 </random>
50 </template>
51 </category>
52
53 <category>
54 <pattern>Quando ^ Blumenau sofreu ^ enchente ^</pattern>
55 <template>
56 <random>
57 <li>Em 2008</li>
58 <li>Blumenau sofreu uma grande enchente em 2008</li>
59 </random>
60 </template>
61 </category>
62
63 <category>
64 <pattern>^ sobre ^ quando Blumenau sofreu ^ enchente ^</pattern>
65 <template>Quando Blumenau sofreu uma grande enchente?</template>
66 </category>
67
68 <category>
69 <pattern>^ Em 2008 ^</pattern>
70 <that>Quando Blumenau sofreu uma grande enchente</that>
71 <template>
72 <random>
73 <li>Isso mesmo! Você está indo muito bem nos estudos!</li>
74 <li>Parabéns, é isso mesmo!</li>
75 <li>Está correto! Parabéns!</li>
76 <li>Acertou em cheio hein!</li>
77 <li>Opa, acertou! Você é fera!</li>
78 <li>Está certo! Desse jeito você vai tirar dez na prova deste
79 assunto!</li>
80 <li>Certíssimo, meus parabéns!</li>
81 </random>
82 </template>
83 </category>
84
85 <category>
86 <pattern>^</pattern>

```

```
87 <that>Quando Blumenau sofreu uma grande enchente</that>
88 <template>
89 <random>
90 <li>Ish, na realidade não é bem isso... Blumenau sofreu uma grande
91 enchente em 2008</li>
92 <li>Ah, não foi dessa vez! Blumenau sofreu uma grande enchente em
93 2008</li>
94 <li>Puxa, não é essa a resposta correta... Blumenau sofreu uma grande
95 enchente em 2008</li>
96 <li>Ops, dessa vez você não acertou. Blumenau sofreu uma grande
97 enchente em 2008</li>
98 <li>Ih, não é isto que está no seu texto! Blumenau sofreu uma grande
99 enchente em 2008</li>
100 </random>
101 </template>
102 </category>
103 ...
```

Fonte: elaborado pela autora.