

CHATTEREDU 2.0: FERRAMENTA WEB DE AUXÍLIO ACADÊMICO UTILIZANDO CHATTERBOT

Eduardo Ferreira Gehrke, Joyce Martins – Orientadora

Curso de Bacharel em Ciência da Computação
Departamento de Sistemas e Computação
Universidade Regional de Blumenau (FURB) – Blumenau, SC – Brazil

efgehrke@furb.br, joyce@furb.br

Resumo: *O presente artigo descreve a extensão do ChatterEDU, um chatterbot gerador de perguntas e respostas a partir do processamento morfossintático de um texto de entrada, desenvolvido por Martins (2016). A principal funcionalidade acrescentada foi a inclusão de opções de resposta incorretas, chamadas de elementos distratores, viabilizando a geração de questões de múltipla escolha. As alternativas erradas são geradas por meio de buscas automatizadas em páginas virtuais, com base na resposta correta. Além disso, o usuário tem a opção de efetuar alterações nas questões antes de persistir as informações e iniciar a conversa com o chatterbot. A aplicação foi migrada para a plataforma web e as questões geradas são sempre persistidas, evitando que estas sejam sobrescritas a cada execução. Foram feitas também melhorias no tratamento de papéis semânticos, corrigindo determinadas inconsistências presentes na primeira versão do ChatterEDU. Ademais, a aplicação gerou perguntas, respostas e distratores adequados para textos de qualquer tema, desde que formados por frases simples. A validação se deu a partir da avaliação das questões geradas, sendo 63% classificadas como de boa qualidade.*

Palavras-chave: *Processamento de linguagem natural. Perguntas de múltipla escolha. Geração de distratores. Chatterbot.*

1 INTRODUÇÃO

Os avanços da ciência e da tecnologia têm sido responsáveis por mudanças em vários segmentos da sociedade, entre os quais o educacional (NASCIMENTO; LUZ; QUELUZ, 2011). Neste cenário, observa-se, como exemplo, a crescente importância que a utilização de ferramentas digitais vem adquirindo, visando auxiliar professores e estudantes no processo de ensino e aprendizagem. Um exemplo de software neste contexto são os *chatterbots*, que, de acordo com Sganderla, Ferrari e Geyer (2003, p. 2), são definidos como "agentes inteligentes desenvolvidos para simular uma conversa através da troca de mensagens de texto, [...] com o intuito de tornar mais familiar a interação entre o homem e a máquina". Com isso, percebe-se que o uso de tais ferramentas na educação pode ser interessante no que diz respeito ao esclarecimento de dúvidas e à recapitulação de conteúdos. Santos, Silva e Brito (2014) também destacam o aproveitamento de *chatterbots* com propósitos educacionais devido à relevância que interfaces baseadas em linguagem natural apresentam atualmente, somada à disponibilidade de acesso à internet. Além disso, Chi et al. (1994 apud LE; KOJIRI; PINKWART, 2011) relatam que o uso de perguntas/respostas no processo de ensino e aprendizagem é benéfico, pois fazer perguntas estimula a auto-explicação.

Neste âmbito, visando disponibilizar um *chatterbot* educacional para possibilitar uma conversação sobre textos da educação básica com suporte em conteúdos de Geografia, Martins (2016) desenvolveu a aplicação desktop ChatterEDU 1.0. O seu funcionamento consiste na inserção de um texto pelo usuário, ocorrendo um processamento por meio do qual são identificados os papéis semânticos, também conhecimentos como temáticos¹, de cada palavra que compõe as sentenças. São gerados, então, pares de perguntas/respostas para uma base de conhecimento em Artificial Intelligence Markup Language (AIML). Assim, torna-se possível que o usuário faça perguntas ao robô ou responda a questionamentos do robô sobre o texto de entrada. A resposta redigida pelo usuário é validada a partir da base de conhecimento previamente gerada. Martins (2016) afirma que um texto para ser processado deve estar gramaticalmente correto e ser composto apenas por sentenças simples com sujeito, verbo e objeto. A autora relata que, mesmo com as restrições estabelecidas, existem sentenças que não são processadas adequadamente, gerando perguntas inconsistentes (incompletas ou contendo parte da resposta), com erros ortográficos ou erros de concordância verbal. Além disso, o usuário deve fazer perguntas e fornecer respostas utilizando a mesma sequência de palavras da base de conhecimento. Por conta disso, mesmo que o usuário responda corretamente uma pergunta, o *chatterbot* pode retornar erro se a resposta não estiver exatamente como gravada na base de conhecimento.

Apesar das limitações descritas, considerando os benefícios que o ChatterEDU 1.0 pode proporcionar, este artigo descreve a migração da ferramenta para a plataforma web, com a inclusão da geração de questões de múltipla

¹ "Papéis temáticos descrevem a relação semântica subjacente entre um verbo e seus argumentos e são usados para descrever padrões léxicos e semânticos no comportamento dos verbos" (KIPPER, 2005 apud SCARTON, 2013, p. 15).

escolha e o aperfeiçoamento do tratamento de papéis semânticos já existentes para evitar gerar perguntas e respostas inconsistentes. Cabe destacar ainda que, no cenário educacional, segundo Papasalouros, Kanaris e Kotis (2008), a aplicação de perguntas de múltipla escolha proporciona a classificação automática de desempenho e feedbacks praticamente imediatos após a conclusão do teste. Além disso, Santos (2015) menciona que a facilidade de acesso à informação permite que atualmente se aprenda em qualquer lugar e a qualquer hora, por meio de dispositivos como computadores e celulares. Considerando-se também a diversidade de ferramentas disponíveis por meio da internet, concebe-se uma importância ao desenvolvimento e disponibilização de aplicações voltadas à educação, com base na tecnologia web.

2 FUNDAMENTAÇÃO TEÓRICA

Esta seção elenca: o assunto que embasa o presente estudo – geração automática de perguntas/respostas com múltipla escolha, a descrição da ferramenta de Martins (2016) e trabalhos correlatos ao ChatterEDU 2.0.

2.1 GERAÇÃO AUTOMÁTICA DE PERGUNTAS/RESPOSTAS COM MÚLTIPLA ESCOLHA

Testes de múltipla escolha oferecem vantagens por abranger vários assuntos em um curto espaço de tempo, possibilitando a avaliação de conhecimentos de maneira rápida. Entretanto, a formulação manual destas questões tende a ser custosa e demandar bastante tempo, sendo observada, neste cenário, a importância de ferramentas que automatizem tal processo (CURTO, 2010).

Alsubait, Parsia e Sattler (2015) destacam a geração automática de perguntas como uma área ainda nova, mas também mencionam o potencial para servir de apoio a educadores em suas atividades. Os autores também conceituam as questões de múltipla escolha através de uma subdivisão em três elementos: (1) uma apresentação do problema ao usuário; (2) a resposta correta; (3) os distratores, que são opções plausíveis de resposta, porém, incorretas.

O processo de geração automática destas questões é resumido por Araki et al. (2016) em duas etapas: (1) geração da pergunta com a sua respectiva resposta correta, a partir de um texto de entrada; (2) geração dos distratores. Al Yahia (2014) elenca três técnicas para geração de perguntas:

- a) baseada em sintaxe: nesta técnica ocorre apenas a análise sintática de cada sentença, conforme descreve Al Yahia (2014). Observa-se que ambiguidades sintáticas, a exemplo da frase "O treinador disse ao auxiliar que sua estratégia seria eficaz.", para a qual mais de uma interpretação é possível, podem gerar perguntas incorretas. Por outro lado, ressalta-se que a técnica pode ser aplicada em textos sobre assuntos diversos. A fim de comparar esta técnica com a baseada em semântica, observa-se que, conforme o Dicionário Michaelis, o termo sintaxe corresponde a "cada um dos elementos da estrutura linguística que determina as relações entre os componentes da oração" (SINTAXE, 2018, p. 1), um conceito menos abrangente do que o descrito na alínea (b);
- b) baseada em semântica: esse tipo de técnica, em geral, conforme descreve Al Yahia (2014), tende a depender do domínio ou do tema tratado, não se adaptando a textos sobre qualquer assunto, como nos outros dois casos. Uma vez que o Dicionário Michaelis define semântica como o "ramo da linguística que estuda a significação das palavras e suas mudanças de sentido ao longo do tempo, bem como a representação do sentido dos enunciados" (SEMÂNTICA, 2018, p.1), nota-se que as perguntas geradas por meio desta técnica tendem a ser mais precisas. Cita-se como exemplo a frase "Alan Turing propôs a máquina de Turing.", cujo processamento do papel semântico TH-humano realizado pelo ChatterEDU 2.0, permite detectar que o verbo "ser" se refere a um ser humano, portanto, a pergunta deve incluir o pronome interrogativo "quem";
- c) baseada em *template*: esta técnica, conforme Liu, Calvo e Rus (2014), apresenta o foco em extrair informações do texto de entrada e estabelecê-las no lugar de espaços vazios em modelos de pergunta, que tendem a ser menos específicas, sugerindo que a técnica possa ser aplicada para contextos mais variados. Além disso, em geral, a complexidade é menor se comparada às outras duas técnicas, que envolvem processamento gramatical. Labutov, Basu e Vanderwende (2015), ao desenvolverem uma aplicação com esta técnica e com base em uma ontologia, ressaltam que as dificuldades proporcionadas ao se criar uma representação semântica foram evitadas.

Vale destacar, ainda, a utilização de ferramentas linguísticas de terceiros para auxiliar em aplicações que envolvem perguntas. Neste contexto, um exemplo é o trabalho de Souza e Moraes (2015), descrito na seção 2.3, que faz uso do *parser* Palavras, também intitulado Visual Interactive Syntax Learning (VISL). Esta ferramenta, que também é utilizada em ambas as versões do ChatterEDU, é responsável por efetuar a anotação morfossintática, que consiste, basicamente, em separar as palavras com suas respectivas classificações morfológicas. Com isso, as palavras categorizadas são analisadas a partir dos elementos mais relevantes da sentença visando à identificação de formatos e, posteriormente, à extração de padrões. Um exemplo abordado pelos autores é a pergunta "Quais são os sintomas da asma?", cujo formato gerado é exposto no Quadro 1.

Quadro 1 - Exemplo de extração de padrão

<p>Formato: <interr>; V @FMV; N @SUBJ>; N</p> <p>Exemplo: [qual] <interr>; [ser] V @FMV; [sintoma] N @SUBJ>; [asma] N @P<</p>
--

Fonte: adaptado de Souza e Moraes (2015).

Assim, com base nos resultados obtidos, viabiliza-se a conversão para AIML. Ocorre, por exemplo, o acréscimo de curingas entre os elementos, simbolizados pelo caractere *. A pergunta e a respectiva resposta para o exemplo anterior, encontra-se em AIML no Quadro 2.

Quadro 2 - Trecho da base de conhecimento AIML gerada

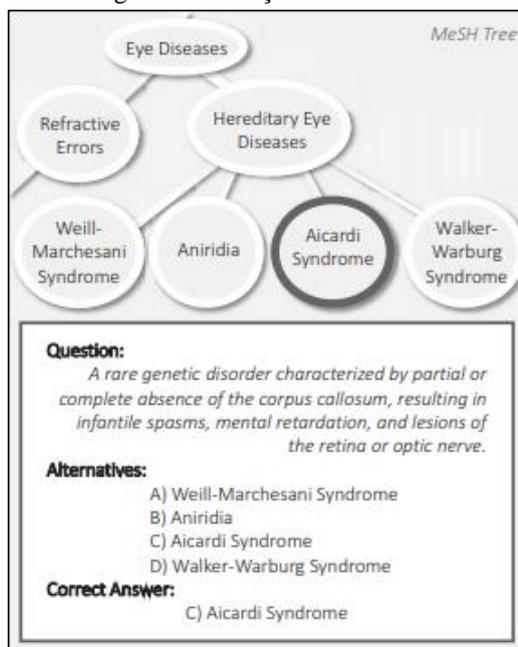
<pre><pattern> QUAL * SINTOMA * ASMA </pattern> <template> Dispneia, tosse e sibilância torácica constituem a tríade clássica de sintomas associados à asma. No entanto, ... </template></pre>

Fonte: adaptado de Souza e Moraes (2015).

Por outro lado, também existem sistemas baseados em *corpus* já pré-classificados, o que evita a necessidade de realizar análises gramaticais do texto de entrada, embora limitem o processamento apenas àquele conteúdo, como ocorre na aplicação de Araki et al. (2016). Neste trabalho, que se baseia em um *corpus* sobre Biologia já semanticamente classificado, são analisadas as relações semânticas existentes em cada frase, para então gerar os questionamentos com base em *templates* pré-definidos.

O método de geração de distratores também varia de acordo com a abordagem proposta. Destaca-se o uso de ontologias, definidas por Al-Yahia (2014, p. 1, tradução nossa) como "estruturas de representação de conhecimento que formalmente descrevem entidades em um domínio e suas relações, possibilitando, assim, inferência e raciocínio automatizados". Alsubait, Parsia e Sattler (2015, p. 1, tradução nossa) ainda acrescentam que "avanços recentes em linguagens e ferramentas de ontologia têm criado um interesse na geração de perguntas de múltipla escolha baseada em ontologia". Este recurso é explorado na abordagem de Lopetegui et al. (2015), que apresentam uma estrutura hierárquica representada por uma árvore gerada após a extração dos conteúdos presentes em um repositório de conceitos médicos. Com isso, a definição dos distratores ocorre por meio da seleção de nodos-irmãos do conceito que corresponde à resposta correta, conforme exemplo exposto na Figura 1.

Figura 1 – Geração de distratores



Fonte: Lopetegui et al. (2015).

Outra estratégia a ser mencionada é o processo adotado por Curto (2010), também descrito na seção 2.3, que consiste em examinar e buscar distratores no próprio texto de entrada. Ocorrem análises de características dos elementos que o compõem, visando identificar semelhanças com a resposta correta e, com base nisto, são definidos potenciais candidatos. Araki et al. (2016) optaram por uma técnica semelhante, descrevendo, inclusive, a formação de

um grafo para retratar os termos-chave contidos no texto e as relações entre eles, sendo então excluído o nodo correspondente à resposta correta para a posterior seleção de distratores.

Por opção de desenvolvimento, o método selecionado para realizar o presente trabalho consistiu em efetuar buscas automatizadas por elementos distratores em páginas virtuais, utilizando a resposta correta como parâmetro. Com isso, a fonte de informações para obtenção de distratores corresponde ao conteúdo dos códigos-fontes das páginas acessadas, fazendo com que a técnica não se equipare especificamente a alguma das outras duas mencionadas.

2.2 CHATTEREDU 1.0

O ChatterEDU 1.0, desenvolvido por Martins (2016), apresentava como objetivo principal a possibilidade de conversar com o usuário sobre textos da educação básica, escritos em língua portuguesa. A aplicação deveria ser disponibilizada por meio de uma plataforma web, com a funcionalidade de formular e responder perguntas, com foco em especial em conteúdos de Geografia. Desenvolvida como uma ferramenta desktop na linguagem Java, o *chatterbot* inclui bases de conhecimento AIML, além de utilizar o Program AB, que as interpreta.

Basicamente, o processamento do texto de entrada realizado por meio da aplicação pode ser descrito através das seguintes etapas:

- inserção do texto de entrada: o usuário digita um texto a ser processado, sobre o qual ele conversará com o robô. Este conteúdo é dividido primeiramente em sentenças e depois em palavras;
- anotação morfossintática (*parser* Palavras): cada sentença é analisada pelo *parser*, que retorna uma lista com as palavras morfologicamente analisadas e seus respectivos papéis semânticos;
- processamento do resultado obtido: com base na classificação semântica das palavras, esta etapa tem a finalidade de obter perguntas (tanto as que podem ser realizadas pelo usuário, quanto pelo robô) e suas respectivas respostas;
- criação de uma base de conhecimento na linguagem AIML: com base nas perguntas e respostas anteriormente obtidas, a base de conhecimento é gerada.

A partir da geração da base de conhecimento, é possível, para o usuário, fazer perguntas ao *chatterbot*, bem como solicitar que ele as faça. No Quadro 3 encontra-se um trecho do código AIML gerado a partir do texto de entrada "Varsóvia é a capital da Polônia."

Quadro 3 – Trecho da base AIML do ChatterEDU 1.0

```
<category>
  <pattern>^ qual ^ é ^ capital ^ Polônia ^</pattern>
  <template>
    <random>
      <li> Varsóvia </li>
      <li> Varsóvia é a capital da Polônia </li>
    </random>
  </template>
</category>
```

Fonte: elaborado pelo autor.

O código representa um padrão (*pattern*) de pergunta que o usuário pode fazer ao robô, sendo que o símbolo ^ indica que podem aparecer nenhuma ou várias palavras. Isto significa que o programa é capaz de interpretar o questionamento "Qual é a capital da Polônia?", como também "Gostaria de saber qual cidade é atualmente a capital da República da Polônia?", por exemplo. É importante ressaltar que a palavra "Qual" foi empregada para a formulação desta pergunta, pois o papel semântico identificado para o vocábulo "Varsóvia" foi o de tema principal da oração (TH), atrelado ao verbo "ser". O Quadro 4 exemplifica os papéis semânticos tratados no trabalho de Martins (2016).

Quadro 4 – Papéis semânticos tratados pelo ChatterEDU 1.0

papel semântico (abreviação)	interrogativa	frase de entrada	pergunta formulada
agente (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?
localização (LOC)	Onde	Santa Catarina fica na região sul.	Onde Santa Catarina fica?
localização temporal (LOC-TMP)	Quando	Blumenau sofreu uma grande enchente em 2008.	Quando Blumenau sofreu uma grande enchente?
origem temporal (ORI-TMP)	Desde quando	Desde 1852 foram registradas 64 enchentes em Blumenau.	Desde quando foram registradas 64 enchentes em Blumenau?
extensão ou quantidade (EXT)	Quanto	O estado mede 95703 quilômetros quadrados.	O estado mede quanto?
extensão de medida temporal (EXT-TMP)	Quanto tempo	A tragédia durou por duas semanas.	A tragédia durou quanto tempo?
tema principal (TH)	O que	Florianópolis tem cerca de 421 mil habitantes.	O que tem cerca de 421 mil habitantes?
tema principal (TH) – verbo "ser"	Qual	Blumenau é a terceira maior cidade de Santa Catarina.	Qual é a terceira maior cidade de Santa Catarina?
tema principal (TH) – humano	Quem	Ele foi empregado em diversas missões.	Quem foi empregado em diversas missões?

Fonte: adaptado de Martins (2016).

Para permitir que o *chatterbot* faça perguntas referentes ao assunto, foi adicionada a palavra "sobre", no início do texto, conforme trecho do código AIML apresentado no Quadro 5.

Quadro 5 - Trecho da base AIML do ChatterEDU 1.0

```
<category>
  <pattern>^ sobre ^ qual ^ é ^ capital ^ Polônia ^</pattern>
  <template> Qual é a capital da Polônia? </template>
</category>
```

Fonte: elaborado pelo autor.

Neste cenário, o usuário pode digitar algo como "Quero que me pergunte sobre qual é a capital da Polônia.", o que fará com que a mensagem do *chatterbot* seja "Qual é a capital da Polônia?". Basta que a resposta seja "Varsóvia", conforme opção listada no Quadro 6, que a afirmação será considerada correta. Será escolhida então, de forma aleatória, uma mensagem entre as relacionadas, expressando um elogio pelo acerto.

Quadro 6 - Trecho da base AIML do ChatterEDU 1.0

```
<category>
  <pattern>^ Varsóvia ^</pattern>
  <that> Qual é a capital da Polônia? </that>
  <template>
    <random>
      <li>Isso mesmo! Você está indo muito bem nos estudos!</li>
      <li>Parabéns, é isso mesmo!</li>
      <li>Está correto! Parabéns!</li>
      <li>Acertou em cheio hein!</li>
      <li>Opa, acertou! Você é fera!</li>
      <li>Está certo! Desse jeito você vai tirar dez na prova deste assunto!</li>
      <li>Certíssimo, meus parabéns!</li>
    </random>
  </template>
</category>
```

Fonte: elaborado pelo autor.

Martins (2016) relata que o trabalho atingiu a maior parte dos objetivos propostos, sendo possível a conversação referente a textos inseridos pelo usuário e posteriormente processados, desde que estejam gramaticalmente corretos. Uma exceção diz respeito à tecnologia web, já que a autora enfrentou problemas durante a implementação, surgindo a necessidade de desenvolver uma aplicação desktop. Houve também menção a possíveis melhorias a serem realizadas, entre as quais: a não sobrescrita da base de conhecimento AIML a cada execução; a inclusão de tratamento de outros papéis semânticos; a adição de um corretor gramatical, permitindo a geração das bases de conhecimento mesmo que o texto não esteja gramaticalmente correto; o desenvolvimento da aplicação para a plataforma web ou móvel; a inclusão de um controle que impeça a repetição de questões já realizadas em um mesmo diálogo; o acréscimo de um mecanismo de aplicação de avaliações a partir da ferramenta; entre outros. Além disso, exemplos de perguntas e

respostas geradas de forma inconsistente estão descritos no Apêndice D, onde são comparados com os resultados obtidos após correções realizadas na presente versão do *chatterbot*.

2.3 TRABALHOS CORRELATOS

A seguir, são explanados três trabalhos identificados como correlatos por possuírem características em comum com o ChatterEDU 2.0, seja por apresentar um modelo de *chatterbot* ou pela geração automática de perguntas/respostas. O primeiro (Quadro 7) é o *chatterbot* web de Brito (2017), que visa responder a perguntas sobre um curso de graduação. Na sequência (Quadro 8), é descrito o *chatbot* de Souza e Moraes (2015), cuja base de conhecimento AIML foi gerada automaticamente. Por último (Quadro 9), é apresentada a aplicação desenvolvida por Curto (2010) para geração automática de perguntas com múltiplas escolhas.

Quadro 7 – *Chatterbot* web com informações sobre um curso de graduação

Referência	Brito (2017)
Objetivos	Desenvolver um <i>chatterbot</i> web para prestar suporte ao sistema de FAQ do curso de Sistemas de Informação da Universidade Federal do Ceará.
Principais funcionalidades	(1) interação entre <i>chatterbot</i> e usuário, visando ao esclarecimento de dúvidas; (2) identificação da intenção do usuário; (3) possibilidade de administrar o conteúdo processado pela aplicação (a exemplo de informações que se alteram com frequência, como o nome do coordenador do curso).
Ferramentas de desenvolvimento	Servidor <i>webhook</i> , Facebook Messenger, IBM Watson, CMS Server, linguagem Python.
Resultados e conclusões	Foram realizados testes, considerados pelo autor como satisfatórios, junto a sete usuários. Com base nos resultados, ocorreram um treinamento e uma atualização da base de dados, alcançando resultados considerados ainda mais positivos em um teste posterior, com um novo grupo de pessoas. O autor menciona vantagens em comparação a um atendente humano, no que diz respeito ao tempo de espera e possibilidade de vários atendimentos simultâneos.

Fonte: elaborado pelo autor.

Quadro 8 – *Chatbot* com base de conhecimento gerada automaticamente

Referência	Souza e Moraes (2015)
Objetivos	Disponibilizar um <i>chatterbot</i> web que conversa sobre doenças, utilizando uma base AIML com informações de uma FAQ sobre o assunto.
Principais funcionalidades	(1) anotação morfosintática dos pares pergunta/resposta extraídos da FAQ, para identificar formatos e extrair padrões; (2) processamento de perguntas do usuário para buscar a resposta na base AIML.
Ferramentas de desenvolvimento	VISL, linguagem AIML, biblioteca JSOUP, plataforma Google App Engine, ferramenta Vaadin.
Resultados e conclusões	Em testes realizados junto a cinco usuários, a facilidade de uso e a naturalidade foram destacadas como pontos positivos, enquanto que o não tratamento de sinônimos e o não processamento de adjetivos foram vistos como aspectos a serem melhorados. Também foi destacado que a ferramenta torna o processo menos custoso, no que diz respeito ao esforço humano aplicado para a construção da base de conhecimento de um <i>chatterbot</i> .

Fonte: elaborado pelo autor.

Quadro 9 – Ferramenta para geração automática de perguntas com múltiplas escolhas

Referência	Curto (2010)
Objetivos	Desenvolver aplicações para gerar testes de múltipla escolha de maneira automática – geração da pergunta, da resposta correta e dos distratores (alternativas incorretas). O autor desenvolveu duas soluções, sendo considerada neste artigo apenas uma delas.
Principais funcionalidades	(1) processamento do texto inserido pelo usuário; (2) análise de um arquivo de especificação de regras, a partir do qual o programa gera as perguntas (o arquivo deve ser escrito e adicionado manualmente pelo usuário); (3) busca, no próprio texto processado, por distratores que possuam características semelhantes às da resposta (por exemplo, classe gramatical, gênero, número); (4) possibilidade do usuário aprovar, editar e descartar cada questão gerada.
Ferramentas de desenvolvimento	Ferramentas de Processamento de Linguagem Natural (PLN) não especificadas pelo autor.
Resultados e conclusões	O autor realizou dois testes, sendo o primeiro com cinco regras e o segundo com vinte e seis. O índice de questões geradas e aprovadas foi, respectivamente, de 77,5% e 68%.

Fonte: elaborado pelo autor.

3 DESCRIÇÃO

Esta seção descreve o desenvolvimento do ChatterEDU na versão 2.0, incluindo os requisitos funcionais e não-funcionais, a especificação e a implementação.

3.1 REQUISITOS

Os Requisitos Funcionais (RF) e os Requisitos Não Funcionais (RNF) definidos para serem atendidos através da atual aplicação encontram-se, respectivamente, nos Quadros 10 e 11. Os casos de uso relacionados aos requisitos funcionais encontram-se detalhados no Apêndice A.

Quadro 10 – Requisitos funcionais

Requisitos funcionais (RF)	Casos de uso (UC)
RF01: o sistema deve possibilitar a interação com o usuário por meio de um chat, respondendo e fazendo perguntas	UC04, UC05, UC06, UC07
RF02: o sistema deve permitir que o usuário forneça como entrada um texto gramaticalmente correto, com sentenças compostas por sujeito, verbo e objeto, para garantir o processamento consistente	UC1
RF03: o sistema deve processar linguisticamente o texto de entrada, para gerar perguntas e respostas consistentes	UC01
RF04: o sistema deve permitir a geração de questões de múltipla escolha	UC01
RF05: o sistema deve acessar e processar páginas web utilizando a resposta como parâmetro, para obter seus elementos distratores	UC01
RF06: o sistema deve disponibilizar a base de conhecimento gerada ao usuário para edições, antes de iniciar o diálogo com o robô	UC03
RF07: o sistema deve permitir que o usuário resete a base de conhecimento, excluindo perguntas persistidas em execuções anteriores	UC02
RF08: o sistema deve permitir que o usuário limpe o histórico da conversa com o robô	UC08

Fonte: elaborado pelo autor.

Quadro 11 – Requisitos não funcionais

Requisitos não funcionais (RNF)
RNF01: o sistema deve apresentar interação com o usuário na língua portuguesa
RNF02: o sistema deve ser desenvolvida na linguagem Java, no ambiente de desenvolvimento NetBeans
RNF03: o sistema deve estar disponível em uma plataforma web
RNF04: o sistema deve operar através do servidor Apache Tomcat por meio do software XAMPP
RNF05: o sistema deve realizar requisições ao site Palavras para efetuar a análise morfológica, obtendo o texto processado para gerar a base de conhecimento AIML
RNF06: o sistema deve extrair informações de páginas HyperText Markup Language (HTML) por meio do <i>parser</i> JSOUP, para obter elementos distratores

Fonte: elaborado pelo autor.

3.2 ESPECIFICAÇÃO

A presente seção, na qual são apresentados detalhes sobre a especificação do projeto, é subdividida em três partes, que expõem, respectivamente, a arquitetura da aplicação e os diagramas de casos de uso e de classes da ferramenta.

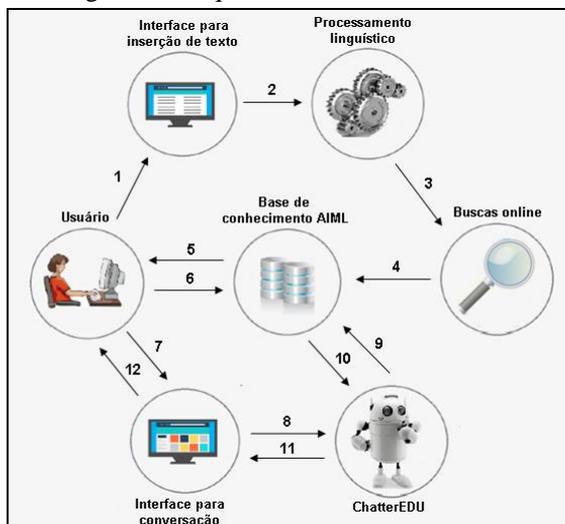
3.2.1 Arquitetura

Ilustrada pela Figura 2, a arquitetura do ChatterEDU 2.0 é composta por seis componentes. Nota-se, em comparação ao ChatterEDU 1.0, a adição do componente *Buscas online*, devido à geração de distratores. Além disso, *Interface para inserção de texto* e *Interface para conversação* foram adaptadas, pois a aplicação passou a funcionar via web. Também ocorreram ajustes em *Processamento linguístico*, através da otimização de tratamentos de papéis semânticos, e em *Base de conhecimento AIML*, visto que esta passou a ser gerada com a inclusão dos distratores, além de não ser mais sobreposta a partir do processamento de cada texto de entrada.

Inicialmente, ocorre a inserção pelo usuário de um texto gramaticalmente correto, com sentenças compostas por sujeito, verbo e objeto (1). Este conteúdo é processado linguisticamente, havendo a geração de perguntas e respostas (2). Com base na resposta de cada uma das perguntas, são realizadas buscas automatizadas nos sites Wikipédia (2018) e Dicionário Criativo (2018), em busca de termos distratores associados à resposta (3). Com isso, os distratores são adicionados às respectivas perguntas e a base de conhecimento AIML é gerada (4). Esta, por sua vez, é disponibilizada ao usuário (5), que tem a opção de alterá-la conforme desejar, podendo editar ou excluir perguntas, respostas e distratores, persistindo-a ao final (6). A partir do momento em que a base está persistida, o usuário utiliza a interface

para conversação para interagir com o *chatterbot* (7). As mensagens por ele digitadas são encaminhadas ao ChatterEDU (8), que, por sua vez, consulta a base de conhecimento AIML (9) para obter a resposta à mensagem (10). Esta é encaminhada à interface para conversação (11), por meio da qual será disponibilizada ao usuário para a continuação do diálogo (12).

Figura 2 – Arquitetura do ChatterEDU 2.0



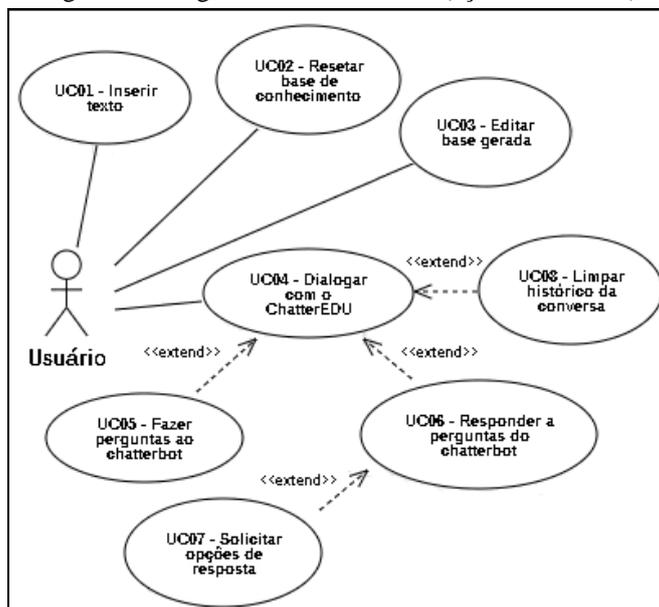
Fonte: elaborado pelo autor.

3.2.2 Diagramas de casos de uso

Houve a adição de quatro novos casos de uso (UC) ao ChatterEDU, em comparação com o trabalho de Martins (2016), conforme ilustra o diagrama presente na Figura 3, desenvolvido em Unified Modeling Language (UML). São representados, no diagrama, os casos de uso de ações do usuário, sendo que os mais significativos também encontram-se explanados no Apêndice A.

A presente versão do *chatterbot* permite que o usuário resete a base de conhecimento, excluindo as perguntas já existentes de persistências anteriores (UC02), podendo também editar o conteúdo AIML gerado após o processamento do texto, excluindo e alterando perguntas, respostas e distratores (UC03). Além disso, o caso de uso *Responder a perguntas do chatterbot* foi estendido, visto que é possível solicitar as opções de resposta (UC07). Por fim, foi incluída a funcionalidade de limpar o histórico da conversa (UC08). Os outros casos de uso representam funcionalidades que já estavam disponíveis anteriormente, correspondendo à inserção do texto pelo usuário para processamento (UC01), além do diálogo com o *chatterbot*, o qual é estendido pelas perguntas feitas pelo usuário e as que ele deve responder (respectivamente UC04, UC05 e UC06).

Figura 3 – Diagrama de casos de uso (ações do usuário)

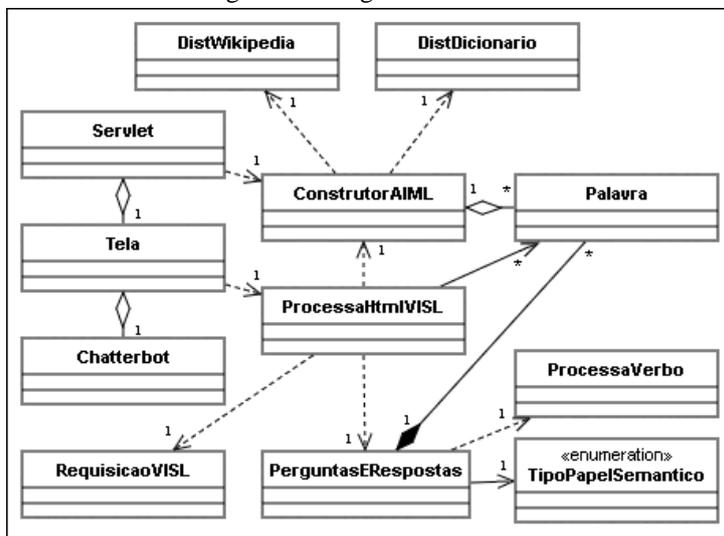


Fonte: elaborado pelo autor.

3.2.3 Diagrama de classes

As classes definidas para o desenvolvimento do ChatterEDU 2.0 estão representadas na Figura 4. Visando obter uma melhor visualização, atributos e métodos foram omitidos nesta figura, porém, o diagrama detalhado encontra-se no Apêndice B.

Figura 4 – Diagrama de classes



Fonte: elaborado pelo autor.

Em comparação à descrição presente no trabalho de Martins (2016), houve a adição de três novas classes. *DistDicionario* e *DistWikipedia* são acionadas pela classe *ConstrutorAIML* para o fornecimento de elementos distratores quando a base de conhecimento está sendo gerada, caracterizando relacionamentos de dependência. Esta classe, por sua vez, foi adaptada visando seguir o padrão Singleton, o que significa que ela é instanciada apenas uma vez a cada execução, facilitando também o acesso às suas funcionalidades a partir de qualquer parte do código, e houve também a inclusão de uma lista de objetos da classe *Palavra*, criando um relacionamento de agregação. A outra classe que foi adicionada é *Servlet*, responsável pelo funcionamento da ferramenta via web, recebendo e enviando informações a um arquivo de extensão Java Server Page (*.jsp*), onde a estrutura em HTML, CSS e JavaScript está disponível. Esta também apresenta um relacionamento de dependência com a classe *ConstrutorAIML*, que precisa ser acionada para atualizações da base de conhecimento.

Ademais, o processamento permanece, de forma geral, semelhante ao do trabalho predecessor. A classe *Tela* implementa as funcionalidades invocadas para o processamento do texto e para conversação, tendo ainda sido acrescentada uma opção para resetar a base de conhecimento. A classe *Chatterbot* viabiliza a utilização dos métodos presentes na biblioteca Program AB para interpretar a base AIML. A classe *RequisicaoVISL* é responsável por enviar o texto de entrada para processamento pelo *parser* *Palavras*, fornecendo o resultado à classe *ProcessaHtmlVISL*, a qual o converte em frases compostas por objetos da classe *Palavra*.

A classe *PerguntasERespostas* possibilita a geração das perguntas para cada frase, conforme identificação dos papéis semânticos nela presentes – processo no qual há auxílio da classe *TipoPapelSemantico*, uma enumeração que contém valores fixos referentes aos papéis. A classe também se comunica com *ProcessaVerbo* para obter ajustes de conjugação verbal quando preciso, e é importante observar que alguns de seus métodos foram ajustados, visando eliminar determinadas inconsistências na geração de perguntas e respostas.

3.3 IMPLEMENTAÇÃO

Esta seção apresenta as técnicas e ferramentas utilizadas na realização do trabalho, descrevendo também como ocorre o processamento do texto. A operacionalidade também é abordada, expondo os requisitos para utilização do ChatterEDU.

3.3.1 Técnicas e ferramentas utilizadas

A implementação da aplicação ocorreu no ambiente de desenvolvimento Netbeans 8.2, na linguagem de programação Java e por meio da tecnologia Java Server Pages. Também foi utilizado o software XAMPP 5.6.21, o qual viabiliza o acesso ao servidor Apache Tomcat. Ainda houve a utilização da biblioteca JSOUP, responsável pela conexão às páginas virtuais e obtenção de seus conteúdos. Mantiveram-se as funcionalidades de geração de base de conhecimento em AIML e a utilização do interpretador Program AB, importado como uma biblioteca no Netbeans, conforme descrito por Martins (2016).

3.3.2 Processamento do texto de entrada

Conforme enfatizado por Martins (2016), a complexidade da aplicação permanece no processamento do texto de entrada, a partir do qual são geradas perguntas e respostas, assunto de uma das três subdivisões desta seção. As outras duas referem-se a funcionalidades adicionais do ChatterEDU 2.0, quais sejam: a geração de elementos distratores para cada resposta e a disponibilização da base AIML gerada para edição pelo usuário, antes de ocorrer a persistência desta.

3.3.2.1 Geração de perguntas e respostas

O método `processaTexto`, da classe `ProcessaHtmlVISL`, continua sendo o responsável pelo processamento do texto de entrada, ocorrendo, a partir deste, a divisão das sentenças em objetos da classe `Palavra` após a classificação de cada vocábulo por meio do `parser` `Palavras`. As requisições a sites continuam sendo feitas pela classe `RequisicaoVISL`, com a alteração de que isto agora é realizado via biblioteca `JSOUP`, e não mais via `HttpCore` e `HttpClient`.

Uma vez obtidas estas sentenças, a classe `PerguntasERespostas` analisa os papéis semânticos presentes em cada uma, com base nos quais são geradas as perguntas e as respostas. É importante destacar que houve melhorias nos métodos desta classe, evitando determinadas inconsistências, conforme descrito na seção 4. As perguntas e respostas geradas são encaminhadas à classe `ConstrutorAIML`, responsável pela geração da base AIML. Antes, no entanto, a classe promove a geração dos distratores.

3.3.2.2 Geração de elementos distratores

A geração de elementos distratores pode ocorrer de duas maneiras: com ou sem numerais, a partir da identificação de alguma palavra com a classe sintática numeral (NUM) na resposta. Também se definiu que para cada pergunta são gerados, na maioria dos casos, oito distratores. Todavia, existem situações excepcionais em que este número pode chegar a dezesseis ou ser inferior. Esta última situação se deve à falta de termos alternativos durante as buscas online, exemplificada pelo vocábulo "paraguaios", para o qual obtém-se apenas "guaranis" e "Demografia do Paraguai".

Quando um numeral é identificado na resposta, ocorre uma geração de números distratores a partir da própria classe `ConstrutorAIML`. Um exemplo é a frase "O Brasil foi descoberto no ano 1500.", que gera a pergunta "Quando o Brasil foi descoberto?". Como distratores para a resposta "No ano de 1500.", são geradas oito opções substituindo o termo "1500" por valores aleatórios entre 10 unidades a menos e 10 unidades a mais, ou seja, entre 1490 e 1510. Este intervalo é sempre aplicado para números identificados entre os valores de 1000 e 2050 devido à maior probabilidade de corresponderem a anos. Para numerais fora deste intervalo, são fornecidos os valores de um décimo, um quarto, um meio, três quartos, cinco quartos, dobro, quádruplo e décuplo, conforme ocorre na sentença "O barril de petróleo vale 80 dólares.", que gera o par pergunta/resposta "O barril de petróleo vale quanto?"/"80 dólares". Neste caso, os distratores fornecidos apresentam a substituição do número 80 na resposta pelos valores 8, 20, 40, 60, 100, 160, 320 e 800. Este tratamento excepcional para numerais se deve ao fato das classes `DistWikipedia` e `DistDicionario` não terem apresentado resultados satisfatórios na busca por distratores para elementos deste tipo.

A segunda forma de se obter distratores ocorre por meio das classes `DistWikipedia` e `DistDicionario`, que realizam buscas automatizadas às páginas virtuais Wikipédia e Dicionário Criativo, respectivamente. Estas classes são acionadas pelo `ConstrutorAIML`, que encaminha a resposta correta como parâmetro e, com base nesta informação, adquire distratores relacionados à resposta. Em ambos os casos, o processamento inicia com a conexão por meio da ferramenta `JSOUP` ao endereço do site tendo como parâmetro a resposta correta, sendo obtida a página HTML que contém informações sobre esta resposta. Na sequência, esta página é analisada para extrair os elementos distratores. O processamento foi desenvolvido com base na nomenclatura das `tags` presentes nas páginas de ambos os sites.

O processamento das páginas do Dicionário Criativo visa localizar a região da página onde se encontram as palavras relacionadas, conforme a Figura 5, que ilustra parte das palavras fornecidas na busca pelo termo "planalto". Nesse caso, um termo-chave no algoritmo é `class="c_primary_hover c_border_hover analogico">`, já que este se encontra nesta área do código-fonte da página, separando os vocábulos, como pode-se observar no Quadro 12, que contém um trecho do código da página da Figura 5.

Figura 5 – Palavras relacionadas ao termo “planalto”



Fonte: adaptado de Planalto (2018).

Quadro 12 – Trecho do código-fonte do Dicionário Criativo

1	<[...]	title="planalto" class="c_primary_hover c_border_hover analogico">
2		Planalto
3		[...]>
4	<[...]	title="outeiro" class="c_primary_hover c_border_hover analogico">
5		Outeiro
6		[...]>
7	<[...]	title="montanha" class="c_primary_hover c_border_hover analogico">
8		Montanha
9		[...]>
10	<[...]	title="serrania" class="c_primary_hover c_border_hover analogico">
11		Serrania
12		[...]>
13	<[...]	title="serra" class="c_primary_hover c_border_hover analogico">
14		Serra
15		[...]>

Fonte: adaptado de Planalto (2018).

O método `split` é executado utilizando o termo-chave como parâmetro, ocorrendo a remoção de *tags* HTML antes e depois de cada um dos elementos, para que os mesmos sejam extraídos. Estes são, então, adicionados a uma lista e selecionados de forma randômica como distratores para a pergunta. É importante ressaltar, ainda, que o primeiro elemento listado na página sempre corresponde à própria resposta, sendo, portanto, removido.

O processamento das páginas da Wikipédia apresenta maior complexidade. Uma vez acessada a página com informações sobre a resposta, são verificadas as categorias presentes no rodapé da página. Analisa-se, inicialmente, se a publicação possui uma categoria de mesmo nome. Esta verificação é ilustrada por meio do Quadro 13, que expõe a situação para a página do município de Blumenau, que possui uma categoria homônima, e para a página da cidade de Campo Erê, que não apresenta.

Quadro 13 – Categorias relacionadas a Blumenau e Campo Erê

Categorias da página do município de Blumenau:	
Categorias: Blumenau Fundações no Brasil em 1850	
Categorias da página do município de Campo Erê:	
Categoria: Municípios de Santa Catarina	

Fonte: adaptado de Wikipédia (2018).

A verificação de categoria homônima à publicação ocorre através da execução do método `contains` utilizando como parâmetro o termo `/wiki/Categoria:` concatenado com o título da própria publicação. Caso este seja encontrado (linha 5 do Quadro 14), a página desta categoria é acessada, para o exemplo de "Blumenau". Já para casos em que isto não se aplica, é realizado então um `split` com o termo `Categoria<a>` ou `Categorias<a>` (linha 2 do Quadro 15), sendo selecionado e acessado o primeiro link após o termo (linha 5 do Quadro 15), para o exemplo de "Campo Erê".

Quadro 14 – Trecho do código-fonte da Wikipédia

1	
2	Categorias:
3	
4	
5	
6	Blumenau
7	
8	
9	<a href="/wiki/Categoria:Funda%C3%A7%C3%B5es no Brasil em 1850" title="
10	Categoria:Fundações no Brasil em 1850">
11	Fundações no Brasil em 1850
12	
12	

Fonte: adaptado de Blumenau (2018).

Quadro 15 – Trecho do código-fonte da Wikipédia

1	
2	Categorias:
3	
4	
5	<a href="/wiki/Categoria:Munic%C3%ADpios de Santa Catarina" title="
6	Categoria:Municípios de Santa Catarina">
7	Municípios de Santa Catarina
7	
8	

Fonte: adaptado de Campo Erê (2018).

Uma vez obtida a página da categoria, ocorre um tratamento que depende do resultado constatado na etapa anterior. Para publicações que não tiveram categoria homônima, verifica-se, mesmo assim, se estas eventualmente tratam do mesmo assunto – isto pode não ter sido detectado anteriormente porque o nome da publicação está no singular e o da categoria no plural. É o que ocorre com a publicação "Planalto", pertencente à categoria "Planaltos". Neste caso, o algoritmo identifica se a página da categoria contém o link para a página da publicação, conforme grifado no Quadro 16, interpretando que a publicação e a categoria tratam, de fato, do mesmo tema.

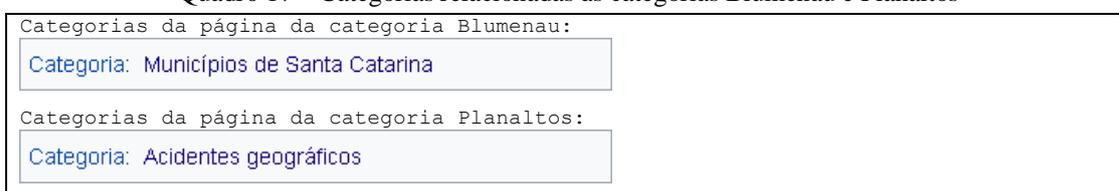
Quadro 16 – Página da categoria "Planaltos"



Fonte: adaptado de Planaltos (2018).

Para esta situação, assim como para publicações com categoria homônima, ocorre uma nova conexão à primeira das categorias identificadas na página atual, o que, no caso da categoria "Blumenau", corresponde a "Municípios de Santa Catarina" e, para "Planaltos", a "Acidentes geográficos", conforme o Quadro 17.

Quadro 17 – Categorias relacionadas às categorias Blumenau e Planaltos



Fonte: adaptado de Wikipédia (2018).

Com isso, a lista de categorias e publicações pertencentes a esta última página acessada é extraída e uma seleção randômica é realizada, em um processo semelhante ao do Dicionário Criativo. Por outro lado, para o exemplo do termo "Campo Erê", a extração ocorre na primeira categoria acessada – neste caso, também "Municípios de Santa Catarina", ao qual se chegou diretamente, conforme ilustrado no Quadro 13.

Para melhor entendimento do processo, uma captura de tela com adaptações da página "Categoria:Municípios de Santa Catarina" é exibida através do Quadro 18.

Quadro 18 – Categoria "Municípios de Santa Catarina"



Fonte: adaptado de Municípios de Santa Catarina (2018).

Vale destacar, ainda, que foram incluídos tratamentos especiais para situações em que a resposta contém mais de uma ocorrência de substantivo, adjetivo e/ou nome próprio, uma vez que a utilização de uma resposta composta dificulta a geração de distratores. Com isso, é necessário identificar um termo específico na resposta para ser

substituído, mantendo o restante inalterado. Verifica-se, primeiramente, a presença de nomes próprios, a exemplo da resposta "A capital do Brasil.", para a qual são buscados apenas termos que substituam "Brasil". Se este não for o caso, analisa-se se a resposta contém um substantivo, seguido de um adjetivo, eventualmente com um artigo à frente deste termo composto. Nesta situação, são gerados oito distratores por meio da substituição do substantivo e outros oito com a troca do adjetivo, sendo um exemplo o termo "Povo indígena", para o qual são gerados distratores como "Povo caipira" e "Cultura indígena". O último caso consiste em averiguar se há mais de um substantivo na resposta, sendo assim substituído o primeiro. Caso nenhum dos casos se aplique à resposta em questão, o que se observa, por exemplo, em respostas com apenas uma palavra, o próprio termo se torna o parâmetro para a busca.

Após esta etapa, estando a base AIML disponível, é executado o método `disponibilizaBase`, que constrói uma String contendo uma estrutura em HTML para possibilitar as edições pelo usuário. Esta é encaminhada à classe `Servlet` e, por meio desta, o conteúdo no arquivo `jsp` é atualizado, expondo, ao usuário, todas as perguntas com a opção de alterá-las.

3.3.2.3 Geração da base AIML

Para a elaboração da base AIML, novos métodos foram desenvolvidos ou adaptados, considerando que o usuário pode solicitar opções de resposta durante a conversa. O Quadro 19 traz uma parte do código da construção da base AIML.

Quadro 19 – Trecho do código para geração da base AIML

```

1 String[] temp = getPerguntasRespostas().toString().split("\n");
2 for (int i = 0; i < temp.length; i++) {
3     [...]
4     aiml.append(getCategoryPerguntaUsuario().toString());
5     aiml.append(getCategoryTemaUsuario().toString());
6     aiml.append(getCategoryRespostaCertaUsuario().toString());
7     aiml.append(getCategoryOpcoesUsuario().toString());
8     aiml.append(getCategoryRespostaCertaOpcoesUsuario().toString());
9     aiml.append(getCategoryRespostaErradaUsuario().toString());
10    aiml.append(getCategoryRespostaErradaOpcoesUsuario().toString());
11    [...]
12 }

```

Fonte: elaborado pelo autor.

Os seguintes métodos já estavam presentes no trabalho de Martins (2016), com as respectivas finalidades:

- `getCategoryPerguntaUsuario`: para quando o usuário faz uma pergunta e o robô responde com a informação da base AIML;
- `getCategoryTemaUsuario`: para quando o usuário solicita e o robô faz uma pergunta sobre determinado assunto da base AIML;
- `getCategoryRespostaCertaUsuario`: para quando o usuário responde à pergunta de maneira correta conforme a base AIML e o robô responde positivamente;
- `getCategoryRespostaErradaUsuario`: para quando o usuário responde de maneira errônea e o robô reage negativamente.

Os outros três métodos foram adicionados para incluir tratamentos para os distratores. `getCategoryRespostaCertaOpcoesUsuario` e `getCategoryRespostaErradaOpcoesUsuario` foram necessários para tratar a resposta do usuário logo após o fornecimento das opções, gerando uma mensagem positiva para caso de acerto ou negativa em caso de erro. Já o método `getCategoryOpcoesUsuario` aborda a situação em que o robô fez a pergunta ao usuário, que escreve qualquer texto contendo o termo "opções". Isto fará com que o *chatterbot* forneça uma relação de alternativas contendo a resposta correta e os distratores, conforme exemplifica o código do Quadro 20.

Quadro 20 – Trecho da base AIML para solicitação de opções

```

<category>
  <pattern>^ opções ^</pattern>
  <that>Qual é a capital da Alemanha|||</that>
  <template>
    <random>
      <li>Londres|||Varsóvia|||Zagreb|||Helsinquia|||Berlim|||</li>
    </random>
  </template>
</category>

```

Fonte: elaborado pelo autor.

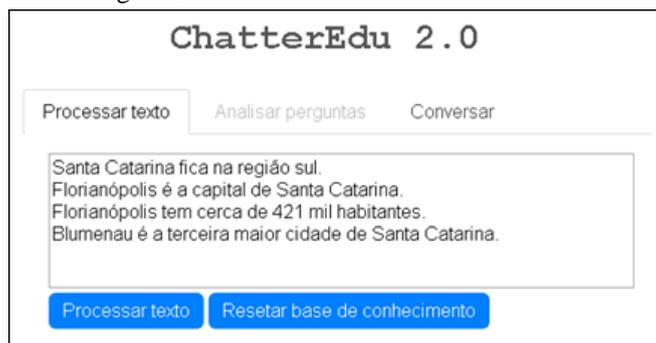
Destaca-se o uso de *pipes* ao final da pergunta, para posterior substituição por ponto de interrogação, e entre os distratores, para identificá-los e separá-los. Ressalta-se também que os distratores são distribuídos de forma randômica, não obedecendo, necessariamente, à ordem na qual estão persistidos na base. Isto foi implementado no método `EnviarMensagem` da classe `Tela`, responsável por processar as respostas do *chatterbot* às mensagens do usuário.

3.3.3 Operacionalidade da implementação

Desenvolvido como aplicativo web, o ChatterEDU 2.0 requer a instalação do software XAMPP ou similar, para viabilizar o acesso ao servidor Apache Tomcat, além de conexão à internet, para possibilitar requisições aos sites consultados durante a execução. O uso da aplicação pode ocorrer por meio de diversos navegadores de internet, a exemplo do Google Chrome, Mozilla Firefox ou Internet Explorer.

A tela inicial é apresentada na Figura 6, na qual o usuário digita o texto que deseja ser processado. Ao clicar em *Processar texto*, inicia-se a geração de perguntas e respostas com base no texto, bem como a busca por distratores, para a posterior criação da base AIML. Se o processamento ocorrer com sucesso, redireciona-se à aba *Analisar perguntas*. Caso contrário, mantém-se na aba atual. O usuário tem ainda a opção *Resetar base de conhecimento*, o que exclui todas as perguntas, respostas e distratores persistidas até aquele momento.

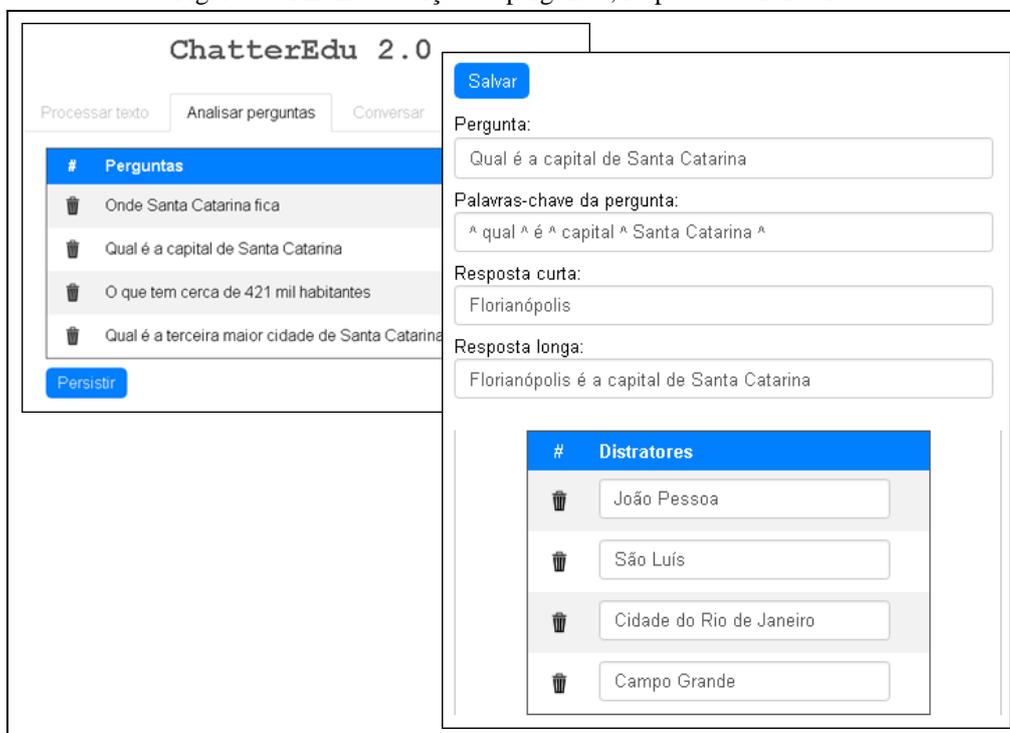
Figura 6 – Processamento do texto de entrada



Fonte: elaborado pelo autor.

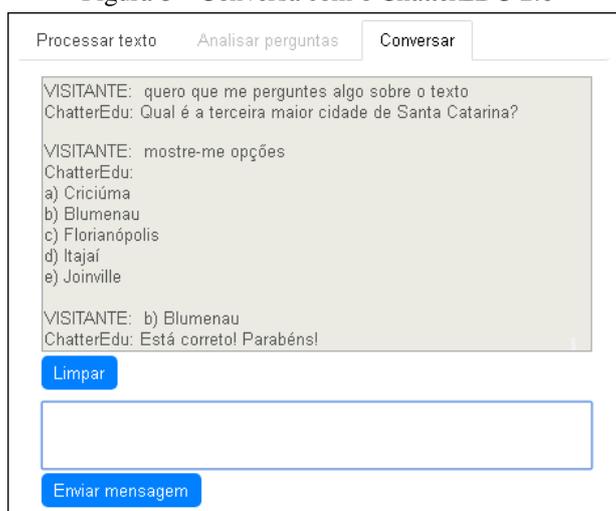
Após o texto de entrada ser processado, o usuário tem a opção de alterar e excluir perguntas, respostas e distratores. Inicialmente, é exibida uma lista com todas as perguntas, conforme a Figura 7 (à esquerda), onde o usuário pode excluí-las ou clicar sobre elas, para obter opções mais detalhadas, de acordo com a Figura 7 (à direita), e realizar as alterações desejadas. Por outro lado, ao selecionar *Persistir*, a base AIML é salva e o usuário é redirecionado para a aba *Conversar*, na qual ocorre o diálogo com o robô, retratado na Figura 8.

Figura 7 – Análise e edição de perguntas, respostas e distratores



Fonte: elaborado pelo autor.

Figura 8 – Conversa com o ChatterEDU 2.0



Fonte: elaborado pelo autor.

Destaca-se ainda, referente à Figura 8, que o usuário tem a opção de apagar todo o histórico da conversa por meio do botão *Limpar*. Além disso, as perguntas também podem ser respondidas diretamente, sem a necessidade de solicitar opções. Por outro lado, para responder uma pergunta após o fornecimento das opções, o usuário pode apenas digitar a letra da alternativa desejada, havendo um processamento para que isto seja interpretado conforme o termo ao qual esta está associada.

4 RESULTADOS

Os objetivos inicialmente propostos para o presente trabalho foram atingidos. A geração de questões de múltipla escolha através de buscas de distratores em páginas virtuais foi implementada, a ferramenta foi migrada para a plataforma web e o tratamento de seis papéis semânticos foi ajustado, evitando gerar perguntas e respostas com algumas inconsistências.

Em relação à geração de distratores, notou-se que o uso de dois sites para a busca das alternativas foi benéfica, uma vez que a Wikipédia se apresentou mais apropriada para fornecer distratores a nomes próprios, a exemplo de pessoas e localidades, enquanto que o Dicionário Criativo demonstrou maior eficácia para substantivos em geral. Um exemplo que ilustra o primeiro caso é um teste com o termo "Varsóvia", em que a Wikipédia retornou as palavras "Madrid", "Oslo", "Valeta" e "Riga", enquanto que o Dicionário Criativo não forneceu resultados para este caso. Por outro lado, quando se buscou o termo "Planalto", o Dicionário Criativo retornou "Outeiro", "Montanha", "Serrania" e "Serra", já a Wikipédia forneceu "Pântanos, charcos e sapais", "Fiordes", "Cordilheiras" e "Cavernas". Observa-se que a Wikipédia retorna os distratores no plural, ou contendo mais de um termo, o que requer adaptações pelo usuário. Portanto, considera-se o Dicionário Criativo mais adequado para esta situação.

Ainda foi possível perceber que, em casos nos quais apenas parte da resposta é extraída para ser substituída, foram comuns problemas de concordância gramatical. Um exemplo é o termo "A capital do Brasil.", no qual "Brasil" é substituído. Com isso, embora a resposta seja consistente, ocorrem inconsistências em caso de trocas por palavras do gênero feminino, tal como nos termos "A capital do Argentina." ou "A capital do Colômbia.". Um processamento para que os artigos fossem adaptados chegou a ser desenvolvido, entretanto, registrou-se uma média de 123 segundos para o processamento de um texto de 41 palavras, valor que foi considerado demasiadamente alto, principalmente ao se analisar os dados apresentados na subseção 4.1. Levando-se em conta que o tratamento ainda apresentava inconsistências, conforme pode ser observado no Apêndice C, e que o usuário tem sempre a opção de alterar os distratores conforme desejar, optou-se por descartá-lo.

Verificou-se, ainda neste contexto, que respostas inconsistentes tendem a gerar distratores inconsistentes, visto que, em muitos casos, apenas uma palavra da resposta é substituída por outro termo. Isto é ilustrado pelo último exemplo no Apêndice E, que foi avaliado como ruim. Embora os distratores substituam o termo "Rússia" por nomes de outros países, a inconsistência se mantém, e a questão permanece ilógica, com a necessidade de maiores adaptações.

Houve dificuldades devido à falta de experiência do autor com tecnologias web, o que, porém, não impediu a migração da ferramenta com êxito para a nova plataforma. Além disso, durante cerca de três semanas ao longo de todo o desenvolvimento, o *parser* Palavras esteve indisponível, o que tornou necessárias alterações no planejamento, visto que atividades dependentes desta ferramenta tiveram de ser interrompidas. Também houve um caso em que a Wikipédia alterou a estrutura de suas páginas, adicionando publicidades de campanhas, o que forçou a adaptações no código para a correta extração das informações. Isto expõe que o funcionamento do *chatterbot* está sempre sujeito à necessidade de

alterações, a partir do momento em que as páginas consultadas apresentarem mudanças em seus conteúdos, ou alterações de nomenclaturas de *tags*.

Além disso, as adaptações no código da classe `PerguntasERespostas` otimizaram a geração de perguntas e respostas e mostraram-se eficazes para evitar parte das inconsistências que vinham ocorrendo. Foram tratados os papéis semânticos AG (voz passiva), LOC, LOC-TMP, ORI-TMP, TH e TH humano, fazendo com que muitas das perguntas geradas requeiram nenhum ou poucos ajustes, conforme observado nos testes realizados e exemplificado no Apêndice D e nos dois primeiros casos do Apêndice E. Houve ainda uma ampliação dos termos tratados no método `ajustesLinguisticos` da classe `PerguntasERespostas`, no qual ocorre a junção de preposições e artigos, uma vez que estes podem ter sido desmembrados durante o processamento linguístico do *parser* `Palavras` para fins de classificação gramatical. Exemplos são as palavras "da", separada em "de" e "a", e "na", desunida em "em" e "a". Um exemplo de correção está relacionado a uma frase que contém o termo "linhas paralelas ao Equador", sendo que esta expressão constava na pergunta gerada como "linhas paralelas a o Equador", com a preposição "a" e o artigo "o" separados. Todavia, após a alteração no código, o termo passou a ser formado corretamente na pergunta.

4.1 TESTES REALIZADOS

A geração de distratores, aliada às melhorias dos processamentos de papéis semânticos, apresentou resultados positivos por meio de testes efetuados com textos de diferentes tamanhos e assuntos. Os conjuntos pergunta/resposta/distratores formulados com base em cada sentença dos textos foram avaliados e classificados conforme uma das três seguintes categorias:

- boa: sem necessidade de adaptações na pergunta nem na resposta, eventualmente com adaptações mínimas nos distratores, a exemplo de ajustes de gênero ou número, tendo gerado ao menos um distrator plausível;
- satisfatória: necessidade de pequena adaptação na pergunta, porém, sem ter comprometido o entendimento da questão. Um exemplo é a substituição do pronome interrogativo por outro mais adequado, ainda com eventuais adaptações na resposta e/ou distratores, tendo gerado ao menos um distrator plausível;
- ruim: necessidade de adaptação mais elaborada na pergunta e/ou resposta e, conseqüentemente, em todos os distratores gerados.

As Tabelas 1 e 2 apresentam, respectivamente, os resultados obtidos em textos de 41 e 78 palavras, divididos nos assuntos de História/Geografia, Informática e Esportes. São listados também os tempos médios de processamento para a geração de perguntas, respostas e distratores, para a conseqüente construção da base AIML. Além disso, consta, no Apêndice E, um exemplo para cada categoria e, no Apêndice F, todos os textos de entrada utilizados nestes testes.

Tabela 1 – Resultados de testes com textos de 5 frases e 41 palavras por assunto

Assunto	Total de perguntas geradas	Perguntas avaliadas como boas	Perguntas avaliadas como satisfatórias	Perguntas avaliadas como ruins	Tempo médio de processamento (s)
História/Geografia	8	6	2	0	33
Informática	8	6	2	0	35
Esportes	10	6	4	0	28
	26	18	8	0	32

Fonte: elaborado pelo autor.

Tabela 2 – Resultados de testes com textos de 9 frases e 78 palavras por assunto

Assunto	Total de perguntas geradas	Perguntas avaliadas como boas	Perguntas avaliadas como satisfatórias	Perguntas avaliadas como ruins	Tempo médio de processamento (s)
História/Geografia	14	8	6	0	78
Informática	14	8	6	0	68
Esportes	14	9	3	2	59
	42	25	15	2	68

Fonte: elaborado pelo autor.

Observa-se, assim como mencionado por Martins (2016), que o processamento do *chatterbot* é aplicável para conteúdos de assuntos diversos, desde que as sentenças do texto de entrada estejam corretas e sejam compostas por frases contendo sujeito, verbo e objeto. Isto se deve ao fato de que aproximadamente 63% das perguntas de ambas as amostras representadas nas Tabelas 1 e 2 foram avaliadas como boas, 34% como satisfatórias e apenas 3% como ruins. Também é importante ressaltar que os testes demonstram uma velocidade média de processamento superior ao dobro da relatada por Martins (2016), que foi equivalente a 15 segundos para o tratamento de um texto com 41 palavras. Além disso, o maior número de perguntas geradas tende a aumentar o tempo de processamento, o que é reforçado por meio da

Tabela 3, na qual são comparados textos do assunto de História/Geografia com o mesmo número de palavras, mas diferente quantidade de frases.

Tabela 3 – Comparação de resultados conforme número de frases em textos de 160 palavras

Número de frases	Total de perguntas geradas	Perguntas avaliadas como boas	Perguntas avaliadas como satisfatórias	Perguntas avaliadas como ruins	Tempo médio de processamento (s)
16	24	16	5	3	107
24	33	23	8	2	166

Fonte: elaborado pelo autor.

Percebe-se, ainda, que o maior número de frases tende a aumentar o tempo de processamento, visto que, para cada sentença tratada, é necessário processá-la semanticamente, o que exige conexões ao site Palavras, ocorrendo também a posterior busca por distratores. É visível também que frases longas tendem a prejudicar o tratamento dos papéis semânticos, gerando um número maior de perguntas avaliadas como ruins.

4.2 COMPARATIVO COM TRABALHOS CORRELATOS

Com o objetivo de fundamentar a importância do presente trabalho, realizou-se um levantamento das características presentes nos trabalhos correlatos descritos na seção 2.3 deste documento. O resultado desta análise está exposto no Quadro 21.

Quadro 21 – Comparativo entre o ChatterEDU 2.0 e os trabalhos correlatos

características correlatos	ChatterEDU 2.0	Brito (2017)	Souza e Moraes (2015)	Curto (2010)
plataforma	web	web	web	não descrito
gera perguntas ou respostas automaticamente a partir de um texto de entrada ou página virtual	perguntas e resposta	respostas	respostas	perguntas e respostas
fornece respostas a perguntas feitas pelo usuário	sim	sim	sim	não
permite que sejam feitas perguntas ao usuário	sim	não	não	sim
gera distratores para as perguntas	sim, a partir de páginas virtuais	não	não	sim, a partir do texto de entrada
é voltado à educação	sim	não	não	sim

Fonte: elaborado pelo autor.

Quanto à plataforma, o ChatterEDU 2.0, bem como as aplicações de Brito (2017) e Souza e Moraes (2015), foram implementadas para web, característica que não é possível afirmar sobre a ferramenta desenvolvida por Curto (2010). Por outro lado, a geração automática de perguntas ou respostas está presente nos quatro trabalhos. Embora Curto (2010) descreva uma etapa manual em sua aplicação (inclusão do arquivo de regras), destaca-se que o pré-processamento de sentenças e a consequente geração de perguntas, respostas e distratores são todos automatizados.

O trabalho de Curto (2010) é o único que não fornece respostas a perguntas feitas pelo usuário, visto que este não consiste em um *chatterbot*. Entretanto, o trabalho apresenta a possibilidade de gerar perguntas para avaliar os conhecimentos do usuário, além de gerar questões de múltipla escolha, assim como ocorre no ChatterEDU 2.0. Porém, é importante destacar que a geração de distratores ocorre com base apenas no texto de entrada, o que tende a gerar resultados menos abrangentes do que por meio de buscas em páginas virtuais, conforme realizado pelo ChatterEDU 2.0. Por último, observa-se que os trabalhos de Brito (2017) e de Souza e Moraes (2015) não são voltados à educação, já que apresentam os respectivos focos de fornecer informações sobre um curso de graduação e de informar o usuário sobre doenças.

Ao comparar as funcionalidades do ChatterEDU 2.0 com as características dos demais, percebe-se que tanto a geração automática de perguntas/respostas a partir de um texto de entrada ou página virtual, quanto a inclusão de perguntas ao usuário, inclusive de múltipla escolha, não são características inéditas quando analisadas individualmente. Porém, o desenvolvimento destas funcionalidades em um *chatterbot*, aliadas ao fornecimento de respostas a perguntas do usuário, representa um diferencial, uma vez que estas funcionalidades não são observadas em conjunto nos trabalhos correlatos descritos.

5 CONCLUSÕES

O presente trabalho consistiu em estender o ChatterEDU 1.0, apresentando, como principal objetivo, incluir a funcionalidade de geração de questões de múltipla escolha. Tal meta foi cumprida por meio de buscas por elementos distratores em páginas virtuais, ocorrendo também a migração da ferramenta para a plataforma web e o aperfeiçoamento do tratamento de papéis semânticos já existentes. Além disso, foram efetuadas melhorias que não estavam planejadas, a exemplo da persistência do conteúdo gerado a cada execução, não sendo mais sobrescrito a cada vez que um texto de entrada é processado, e a opção de editar a base de conhecimento gerada antes de iniciar a conversação com o robô.

Inicialmente, a meta de gerar questões de múltipla escolha se limitava ao assunto de Geografia de ensino básico, entretanto, no decorrer da elaboração do ChatterEDU 2.0, os resultados atingidos indicavam que o funcionamento seria aplicável para textos de qualquer assunto. Desta forma, o objetivo do trabalho foi ampliado, não considerando mais apenas o tema previamente proposto. É importante destacar que permanece a necessidade de o texto ser gramaticalmente correto, contendo sentenças que obedeçam à ordem sujeito, verbo e objeto, para um desempenho preciso. Além disso, ao se tentar aperfeiçoar processamentos para evitar incoerências gramaticais em distratores, houve aumento no tempo de processamento, conforme descrito anteriormente.

Para as buscas automatizadas por distratores em páginas virtuais, foram testadas outras ferramentas alternativas às bibliotecas `HttpCore` e `HttpClient`, utilizadas no ChatterEDU 1.0. A biblioteca `JSOUP` foi considerada a mais apropriada, apresentando um menor número de falhas e uma documentação detalhada, o que foi importante devido à falta de experiência do autor com este tipo de aplicação. Algumas dificuldades ocorreram ao migrar o *chatterbot* para a plataforma web e ao desenvolver códigos em JavaScript, para os quais foi necessário um estudo mais aprofundado destas tecnologias para elaborar as funcionalidades.

Em relação às contribuições deste trabalho, destacam-se os pontos mencionados por Martins (2016), que enfatiza a geração dinâmica da base de conhecimento AIML, em contrapartida aos *chatterbots* que são baseados em conteúdos fixos. Isto se aplica de forma semelhante à geração dos distratores, que, na maior parte dos trabalhos consultados, se baseia em uma ontologia pré-disponibilizada ou no próprio texto de entrada. Ainda, como também mencionado pela autora, a complexidade da língua portuguesa é um tópico relevante neste contexto, sendo percebida, no presente trabalho, em inconsistências gramaticais em distratores, considerando a não-concordância de artigos e adjetivos. Acredita-se que tal dificuldade seria menos marcante em idiomas que possuem estruturas mais simples, a exemplo do inglês. Além disso, é importante frisar o pequeno número de publicações em língua portuguesa que abordem a geração automática de questões de múltipla escolha, fato reforçado pela ausência de referências no idioma neste artigo.

Por fim, conforme descrito por Martins (2016), este trabalho representa uma contribuição para a área de educação, podendo ser usado para auxiliar professores e estudantes em avaliações. Ademais, é interessante citar uma contribuição que a ferramenta apresenta para a área de entretenimento, uma vez que questões de múltipla escolha são comuns em programas de perguntas e respostas, nos quais os conhecimentos dos participantes são testados em um contexto competitivo.

5.1 EXTENSÕES

Com o cumprimento dos objetivos propostos para o presente trabalho, destacam-se as seguintes melhorias que podem ser realizadas em projetos futuros, além das já mencionadas por Martins (2016) e não atendidas neste trabalho:

- a) hospedar a aplicação em um domínio, disponibilizando-a para um maior número de usuários;
- b) estruturar a arquitetura do ChatterEDU, definindo uma camada intermediária que viabilize o processamento de vários idiomas, para evitar que todo o *chatterbot* precise ser reformulado;
- c) desenvolver tratamentos em outros idiomas, uma vez que o *parser* Palavras oferece processamentos em várias línguas;
- d) definir papéis para usuários (um administrador, responsável por manter a base de conhecimento, e um estudante, que dialoga com o robô);
- e) aperfeiçoar os tratamentos de papéis semânticos já existentes, a fim de reduzir as inconsistências remanescentes;
- f) persistir a base AIML através de um Sistema Gerenciador de Banco de Dados (SGBD), ao invés de um arquivo de texto;
- g) reduzir o tempo de busca por distratores por meio de novas técnicas, otimizando o desempenho da aplicação;
- h) incluir o processamento para frases com outras estruturas gramaticais, ou seja, que não estejam na ordem sujeito, verbo e objeto;
- i) gerar uma ontologia de distratores a partir do processamento linguístico.

REFERÊNCIAS

- AL-YAHIA, M. Ontology-based multiple choice question generation. **The Scientific World Journal**, [New York], v. 2014, p. 1-9, Mar. 2014. Disponível em: <<https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3984768/pdf/TSWJ2014-274949.pdf>>. Acesso em: 07 maio 2018.
- ALSUBAIT, T.; PARSIA, B.; SATTTLER, U. Generating multiple choice questions from ontologies: how far can we go? In: LAMBRIX, P. et al. (Eds.) **Knowledge engineering and knowledge management**. Cham: Springer, 2015. p. 66-79. Disponível em: <https://link.springer.com/chapter/10.1007/978-3-319-17966-7_7>. Acesso em: 07 maio 2018.
- ARAKI, J. et al. Generating questions and multiple-choice answers using semantic analysis of texts. In: INTERNATIONAL CONFERENCE ON COMPUTATIONAL LINGUISTICS, 26., 2016, Osaka. **Anais...** Osaka: Association for Natural Language Processing, 2016. p. 1125-1136. Disponível em: <<http://aclweb.org/anthology/C16-1107>>. Acesso em: 18 mar. 2018.
- BLUMENAU. In: WIKIPÉDIA, 2018. Disponível em <<https://pt.wikipedia.org/wiki/Blumenau>>. Acesso em 21 nov. 2018.
- BRITO, F. N. **Desenvolvimento de um chatterbot para a página web de um curso de nível superior**. 2017. 48 f. Trabalho de Conclusão de Curso (Bacharelado em Engenharia de Software) - Universidade Federal do Ceará, Quixadá. Disponível em: <<http://www.repositoriobib.ufc.br/00003a/00003acc.pdf>>. Acesso em: 22 mar. 2018.
- CAMPO ERÊ. In: WIKIPÉDIA, 2018. Disponível em <https://pt.wikipedia.org/wiki/Campo_Erê>. Acesso em 21 nov. 2018.
- CURTO, S. S. L. **Geração automática de testes de escolha múltipla**. 2010. 66 f. Dissertação (Mestrado em Sistemas de Informação e Engenharia da Computação) –Universidade Técnica de Lisboa, Lisboa. Disponível em: <<http://www.inesc-id.pt/publications/4117/pdf>>. Acesso em: 11 mar. 2018.
- DICIONÁRIO CRIATIVO. **Dicionário Criativo**: beta v 3.0. [S.l.], 2018. Disponível em: <<https://dicionariocriativo.com.br/>>. Acesso em: 14 nov. 2018.
- LABUTOV, I; BASU, S.; VANDERWENDE, L. Deep questions without deep understanding. In: ANNUAL MEETING OF THE ASSOCIATION FOR COMPUTATIONAL LINGUISTICS / INTERNATIONAL JOINT CONFERENCE ON NATURAL LANGUAGE PROCESSING, 53rd / 7th, 2015, Beijing. **Proceedings...** Beijing: Association for Computational Linguistics, 2015. p. 889-898. Disponível em: <<http://www.aclweb.org/anthology/P15-1086>>. Acesso em: 14 nov. 2018.
- LE, N. T.; KOJIRI, T.; PINKWART, N. Automatic question generation for educational applications: the state of art. In: van DO, T.; THI, H.; NGUYEN, N. (Eds.) **Advanced computational methods for knowledge engineering**. Berlin: Springer, 2011. p. 325-338. Disponível em: <https://link.springer.com/chapter/10.1007/978-3-319-06569-4_24>. Acesso em: 16 mar. 2018.
- LIU, M.; CALVO, R. A.; RUS, V. Automatic generation and ranking of questions for critical review. **Education Technology & Society**, [Athabasca], v. 17, n. 2, p. 333-346, 2014. Disponível em: <<https://pdfs.semanticscholar.org/1a16/9d0d0e9a41aebeca75a15aea242e9105d976.pdf>>. Acesso em: 09 nov. 2018.
- LOPETEGUI, M. et al. A Novel multiple choice question generation strategy: alternative uses for controlled vocabulary thesauri in biomedical-sciences education. In: AMIA ANNUAL SYMPOSIUM PROCEEDINGS, 2015, [S.l.]. **Proceedings...** Bethesda: American Medical Informatics Association, 2015. p. 861–869. Disponível em: <<https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4765657/pdf/2236850.pdf>>. Acesso em: 07 maio 2018.
- MARTINS, C. **Ferramenta de auxílio acadêmico utilizando chatterbot**. 2016. 50 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/2016_2_camila-viviani-martins_monografia.pdf>. Acesso em: 02 fev. 2018.
- MUNICÍPIOS de Santa Catarina. In: WIKIPÉDIA, 2018. Disponível em <https://pt.wikipedia.org/wiki/Categoria:Munic%C3%ADpios_de_Santa_Catarina>. Acesso em 21 nov. 2018.
- NASCIMENTO, D. E.; LUZ, N. S.; QUELUZ, M. **Tecnologia e sociedade**: transformações sociais. Curitiba: UTFPR, 2011.
- PAPASALOUROS, A.; KANARIS, K.; KOTIS, K. Automatic generation of multiple choice questions from domain ontologies. In: IADIS INTERNATIONAL CONFERENCE E-LEARNING, 1., 2008, Amsterdam. **Proceedings...** [S.l.]: IADIS Press, 2008. p. 427-434. Disponível em: <<https://pdfs.semanticscholar.org/9892/cd189392c491867d5e6d22dcb02d1019923.pdf>>. Acesso em: 07 maio 2018.
- PLANALTO. In: DICIONÁRIO Criativo, 2018. Disponível em <<https://dicionariocriativo.com.br/planalto>>. Acesso em 21 nov. 2018.
- PLANALTOS. In: WIKIPÉDIA, 2018. Disponível em <<https://pt.wikipedia.org/wiki/Categoria:Planaltos>>. Acesso em 21 nov. 2018.
- SANTOS, D. R.; SILVA, F. S.; BRITO, A. J. Uma abordagem para construção de chatterbots educacionais. In: CONFERÊNCIA INTERNACIONAL SOBRE INFORMÁTICA NA EDUCAÇÃO, 19., 2014, Fortaleza. **Anais...** Fortaleza: Universidade Federal do Ceará, 2014. p. 748-751. Disponível em: <http://www.tise.cl/volumen10/TISE2014/tise2014_submission_245.pdf>. Acesso em: 02 fev. 2018.
- SANTOS, G. de S. Espaços de aprendizagem. In: BACICH, L.; TANZI NETO, A.; TREVISANI, F. M (Org.). **Ensino híbrido**: personalização e tecnologia na educação. Porto Alegre: Penso, 2015.

SCARTON, C. 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: 13 dez. 2018.

SEMÂNTICA. In: DICIONÁRIO Michaelis. Editora Melhoramentos, 2018. Disponível em <<https://michaelis.uol.com.br/moderno-portugues/busca/portugues-brasileiro/semantica/>>. Acesso em 21 nov. 2018.

SGANDERLA, R. B.; FERRARI, D. N.; GEYER, C. F. R. BonoBOT: um chatterbot para interação com usuários em um sistema tutor inteligente. In: SIMPÓSIO BRASILEIRO DE INFORMÁTICA NA EDUCAÇÃO, 14., 2003, Rio de Janeiro. **Anais...** Rio de Janeiro: NCE-UFRJ, 2003. Não paginado. Disponível em: <<http://www.nce.ufrj.br/sbie2003/publicacoes/paper46.pdf>>. Acesso em: 28 jan. 2018.

SINTAXE. In: DICIONÁRIO Michaelis. Editora Melhoramentos, 2018. Disponível em <<https://michaelis.uol.com.br/moderno-portugues/busca/portugues-brasileiro/SINTAXE/>>. Acesso em 21 nov. 2018.

SOUZA, L. S.; MORAES, S. M. W. Construção automática de uma base AIML para Chatbot: um estudo baseado na extração de informações a partir de FAQs. In: ENCONTRO NACIONAL DE INTELIGÊNCIA ARTIFICIAL E COMPUTACIONAL, 12., 2015, Natal. **Anais...** Natal: UFRN. Não paginado. Disponível em: <<http://www.lbd.dcc.ufmg.br/colecoes/eniac/2015/021.pdf>>. Acesso em: 11 mar. 2018.

WIKIPÉDIA. **Bem vindos à Wikipédia**. 2018. Disponível em <https://pt.wikipedia.org/wiki/Wikip%C3%A9dia:P%C3%A1gina_principal>. Acesso em 14 nov. 2018.

APÊNDICE A – DETALHAMENTO DOS CASOS DE USO

São apresentadas, nos Quadros 22, 23 e 24, informações detalhadas dos casos de uso UC01 – Inserir texto, UC03 – Editar base gerada e UC04 – Dialogar com o ChatterEDU.

Quadro 22 – Caso de uso UC01 – Inserir texto

UC01 – Inserir texto: permite que o usuário insira um texto sobre o qual quer conversar com o ChatterEDU	
Descrição	Permite inserir um texto sobre qualquer assunto
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 para gerar perguntas e respostas, e realiza buscas em páginas virtuais para obter distratores. 3. O protótipo cria a base AIML a partir do processamento realizado. 4. O protótipo redireciona o usuário para editar as perguntas, respostas e distratores gerados.
Pré-condições	O texto inserido não deve conter erros gramaticais e cada frase deve conter sujeito, verbo e objeto, nessa ordem.
Pós-condições	Base AIML gerada e disponibilizada para edição.

Fonte: elaborado pelo autor com base em Martins (2016).

Quadro 23 – Caso de uso UC03 – Editar base gerada

UC03 – Editar base gerada: permite que o usuário altere e exclua perguntas, respostas e distratores gerados pelo ChatterEDU, podendo persistir o conteúdo e iniciar a conversação em seguida	
Descrição	Permite editar a base AIML gerada pelo ChatterEDU
Ator	Usuário
Cenário principal	<ol style="list-style-type: none"> 1. O usuário analisa as perguntas, respostas e distratores geradas pelo ChatterEDU. 2. O usuário realiza as alterações e exclusões conforme desejar. 3. O usuário seleciona a opção de persistir a base. 4. O protótipo persiste a base AIML. 5. O protótipo redireciona o usuário para iniciar a conversação.
Pré-condições	O texto inserido precisa ter gerado no mínimo uma pergunta.
Pós-condições	Base AIML persistida e usuário redirecionado para conversação com o ChatterEDU.

Fonte: elaborado pelo autor.

Quadro 24 – Caso de uso UC04 – Dialogar com o ChatterEDU

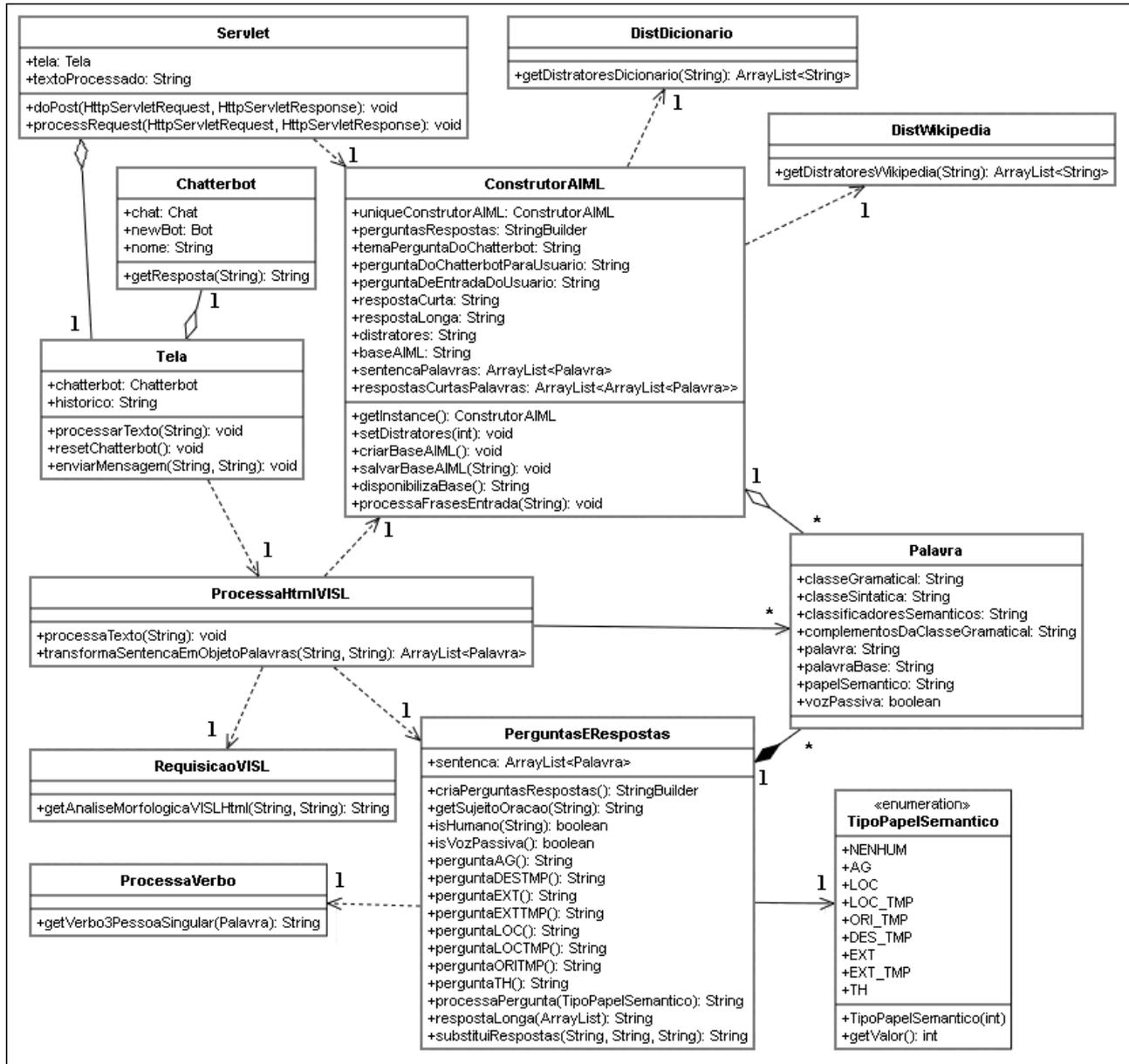
UC04 – Dialogar com o ChatterEDU: permite que o usuário faça perguntas ao robô ou responda a perguntas elaboradas pelo ChatterEDU	
Descrição	Permite ao usuário fazer ou responder a perguntas do ChatterEDU
Ator	Usuário
Cenário principal	<ol style="list-style-type: none"> 1. O usuário digita as saudações iniciais. 2. O usuário escreve sobre qual tema gostaria de ser perguntado. 3. O robô faz a pergunta ao usuário. 4. O usuário solicita que o robô forneça as opções de resposta. 5. As opções de resposta são apresentadas pelo ChatterEDU. 6. O usuário responde à pergunta ao <i>chatterbot</i>. 7. ChatterEDU retorna a informação de que a pergunta foi ou não respondida corretamente.
Fluxo alternativo 01	<ol style="list-style-type: none"> 1. O usuário escreve 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.
Fluxo alternativo 02	<ol style="list-style-type: none"> 1. O usuário faz uma pergunta sobre o texto ao <i>chatterbot</i>. 2. ChatterEDU fornece uma resposta à pergunta formulada.
Pré-condições	<p>Disponer de uma base AIML a partir de UC01 e UC03.</p> <p>A conversação não deve conter erros ortográficos ou gramaticais.</p>
Pós-condições	Histórico da conversação apresentado.

Fonte: elaborado pelo autor com base em Martins (2016).

APÊNDICE B – DETALHAMENTO DO DIAGRAMA DE CLASSES

A Figura 9 ilustra uma versão mais detalhada do diagrama de classes. Com o fim de obter uma visualização mais adequada, foram mantidos apenas os métodos mais relevantes, sendo omitidos, por exemplo, os construtores e outros menos significativos. Há destaque para a classe `ConstrutorAIML`, com seus métodos e atributos característicos do padrão Singleton.

Figura 9 – Diagrama de classes detalhado



Fonte: elaborado pelo autor.

APÊNDICE C – EXEMPLO DE AJUSTES EM DISTRADORES

O Quadro 25 expõe ajustes de número e gênero em artigos presentes em distratores. Todavia, devido às inconsistências e ao aumento do tempo de processamento, tais ajustes não foram mantidos.

Quadro 25 – Exemplo de ajuste de artigos em distratores

O que é um formalismo para a representação de algoritmos?
Uma lei universal
Uma reprodutibilidade universal
Umas neurociências Forenses universal
Uma aparência universal
Umo veículo universal
Uma viatura universal
Umo transporte universal
Uma condução universal
Uma máquina universal

Fonte: elaborado pelo autor

APÊNDICE D – EXEMPLOS DE MELHORIAS NO TRATAMENTO DE PAPÉIS SEMÂNTICOS

No Quadro 26 tem-se a descrição das melhorias realizadas em 6 dos 10 tratamentos de papéis semânticos, incluindo o tratamento efetuado e o comparativo dos resultados do ChatterEDU 2.0 com a versão anterior.

Quadro 26 – Melhorias no tratamento de papéis semânticos

<p>Papel semântico: LOC-TMP Melhoria: parte da resposta aparecia na própria pergunta em frases contendo termos compostos, o que foi corrigido.</p> <p><u>Sentença: Um tsunami devastou a costa do Japão em 2011.</u> Processamento do ChatterEDU 1.0: - Pergunta: Quando um tsunami devastou a costa do Japão em 2011 - Palavras-chave: Quando ^ tsunami devastou ^ costa ^ - Resposta: Do em 2011 Processamento do ChatterEDU 2.0: - Pergunta: Quando um tsunami devastou a costa do Japão - Palavras-chave: Quando ^ tsunami devastou ^ costa ^ - Resposta: Em 2011</p> <p><u>Sentença: O Brasil foi descoberto no mês de abril.</u> Processamento do ChatterEDU 1.0: - Pergunta: Quando o Brasil foi descoberto abril - Palavras-chave: Quando ^ Brasil ^ descoberto ^ - Resposta: No mês de Processamento do ChatterEDU 2.0: - Pergunta: Quando o Brasil foi descoberto - Palavras-chave: Quando ^ Brasil ^ descoberto ^ - Resposta: No mês de abril</p> <p>Papel semântico: LOC Melhoria: acréscimo da partícula "se" em perguntas geradas com o sujeito indeterminado e inclusão de advérbios na pergunta, quando mais apropriado, ao invés de na resposta.</p> <p><u>Sentença: A maioria das favelas não dispõe de infraestrutura básica.</u> Processamento do ChatterEDU 1.0: - Pergunta: Onde dispõe de infraestrutura básica - Palavras-chave: onde ^ dispõe ^ infraestrutura - Resposta: A maioria de as favelas não Processamento do ChatterEDU 2.0: - Pergunta: Onde não se dispõe de infraestrutura básica - Palavras-chave: Onde ^ ^ não ^ dispõe ^ infraestrutura ^ - Resposta: Na maioria das favelas</p> <p>Papel semântico: AG (voz passiva) Melhoria: correções na conjugação do verbo ser, que era incluído erroneamente no singular e no passado.</p>
--

Sentença: As Ilhas Malvinas são administradas pelo governo britânico.

Processamento do ChatterEDU 1.0:

- Pergunta: As Ilhas Malvinas era administradas por quem
- Palavras-chave: ^ quem ^ Ilhas Malvinas ^ administradas ^ ^
- Resposta: Pelo governo britânico

Processamento do ChatterEDU 2.0:

- Pergunta: As Ilhas Malvinas são administradas por quem
- Palavras-chave: ^ quem ^ Ilhas Malvinas ^ administradas ^ ^
- Resposta: Pelo governo britânico

Papel semântico: TH

Melhoria: correções na conjugação dos verbos, que eram incluídos erroneamente no singular.

Sentença: As terras deram lucros aos proprietários.

Processamento do ChatterEDU 1.0:

- Pergunta: As terras deu o que
- Palavras-chave: ^ o que ^ terras deu ^ ^
- Resposta: Lucros aos proprietários

Processamento do ChatterEDU 2.0:

- Pergunta: As terras deram o que
- Palavras-chave: ^ o que ^ terras deram ^ ^
- Resposta: Lucros aos proprietários

Papel semântico: TH (humano)

Melhoria: correções em casos contendo o verbo ser, que era conjugado erroneamente, sendo incluído no singular e no passado, e do pronome interrogativo, adaptado de "qual" para "quem".

Sentença: Os militares foram empregados em diversas missões para auxiliar na tragédia.

Processamento do ChatterEDU 1.0:

- Pergunta: Qual foi empregados em diversas missões para auxiliar na tragédia
- Palavras-chave: ^ qual ^ foi empregados ^ missões ^ auxiliar ^ tragédia ^
- Resposta: os militares

Processamento do ChatterEDU 2.0:

- Pergunta: Quem foram empregados em diversas missões para auxiliar na tragédia
- Palavras-chave: ^ quem ^ foram empregados ^ missões ^ auxiliar ^ tragédia ^
- Resposta: os militares

Papel semântico: ORI-TMP

Melhoria: o termo "desde" não era incluído nas palavras-chave da pergunta por estar contido na resposta. Entretanto, por ser um termo importante neste papel semântico, foi incluído.

Sentença: Blumenau registrou 64 enchentes desde 1852.

Processamento do ChatterEDU 1.0:

- Pergunta: Blumenau registrou 64 enchentes desde quando
- Palavras-chave: ^ quando ^ Blumenau registrou ^ enchentes ^
- Resposta: Desde 1852

Processamento do ChatterEDU 2.0:

- Pergunta: Blumenau registrou 64 enchentes desde quando
- Palavras-chave: ^ desde ^ quando ^ Blumenau registrou ^ enchentes ^
- Resposta: Desde 1852

Fonte: elaborado pelo autor

APÊNDICE E – EXEMPLOS DE PERGUNTAS AVALIADAS

Por meio do Quadro 27, são exemplificadas cada tipo de categoria identificada durante os testes de geração de perguntas/resposta/distratores: uma avaliada como boa, uma como satisfatória e uma como ruim.

Quadro 27 – Exemplos de perguntas conforme avaliações nos testes

<p>Pergunta avaliada como boa, sendo necessários apenas ajustes nos distratores, conforme gênero ou número: Quem possui 27 unidades federativas? O Chile O Suriname O Venezuela O Equador O Brasil</p> <p>Pergunta avaliada como satisfatória, sendo necessário ajuste do pronome interrogativo, além de gênero ou número em distratores: Qual é Estado-membro da União Europeia desde 2004? A Noruega A Liechtenstein A Áustria A Suíça A Eslováquia</p> <p>Pergunta avaliada como ruim, sendo necessárias adaptações mais elaboradas na pergunta, na resposta e em todos os distratores: Onde a Rússia faz? A maior de suas fronteiras com a Camboja A maior de suas fronteiras com a Maldivas A maior de suas fronteiras com a Egito A maior de suas fronteiras com a Timor-Leste A maior de suas fronteiras com a Mongólia</p>

Fonte: elaborado pelo autor

APÊNDICE F – TEXTOS UTILIZADOS NOS TESTES

No Quadro 28 estão disponíveis os textos utilizados durante a fase de testes da aplicação, divididos conforme assunto e tamanho.

Quadro 28 – Textos utilizados nos testes

<p>Geografia/História (41 palavras): O Brasil segue a divisão regional estabelecida em 1970. Santa Catarina é habitada por barrigas-verdes. Porto Alegre é a principal cidade do Rio Grande do Sul. Blumenau é a terceira maior cidade de Santa Catarina. Blumenau registrou 64 enchentes desde 1852.</p> <p>Informática (41 palavras): Richard Bird propôs a máquina NORMA. A máquina de Turing foi proposta em 1936. A Apple é a maior empresa de tecnologia do mundo. Um SGBD disponibiliza interfaces de administração de bases de dados. Chaves estrangeiras descrevem relacionamentos entre tabelas distintas.</p> <p>Esportes (41 palavras): Times de basquetebol são formados por cinco jogadores. O Palmeiras é o maior campeão do Brasil. A Alemanha venceu a Copa em 2014. O Clube Atlético Metropolitano é o principal time de Blumenau. Geninho é o treinador do Avaí desde 2017.</p> <p>Geografia/História (78 palavras): Islamabad é a capital do Paquistão. O Japão é um arquipélago no Oceano Pacífico. A Estônia é o principal parceiro comercial da Finlândia. A costa polonesa é banhada pelo Mar Báltico.</p>
--

A Rússia possui a maior extensão territorial do mundo.
A China possui a maior população do mundo.
Helsinque é a capital e maior cidade da Finlândia.
A Angola é o país lusófono africano de maior população.
A bandeira da Suíça tem uma cruz branca com fundo vermelho.

Informática (78 palavras):

Assembly é a linguagem mais próxima do código de máquina.
Compiladores traduzem programas em linguagem de máquina.
Um código-fonte contém instruções em uma linguagem de programação.
A linguagem de programação mais difícil é Malbolge.
A linguagem Pascal foi criada em 1970.
O algoritmo de Fleury identifica ciclos eulerianos em grafos.
O padrão Singleton garante a existência de instâncias únicas de classes.
A camada de aplicação é a sétima no modelo OSI.
Um backup é uma cópia de segurança.

Esportes (78 palavras):

Um time de futebol é composto por 11 jogadores.
O Figueirense foi fundado em 1921.
O curling é um esporte de inverno popular no Canadá.
O basquetebol foi inventado nos Estados Unidos.
Praticantes de handebol utilizam uma cola especial nas mãos.
A seleção argentina não conquista títulos desde 1993.
O Fortaleza esteve na Série C durante oito anos.
Roger Federer é o maior tenista de todos os tempos.
O Signal Iduna Park é o maior estádio da Alemanha.

Geografia/História (16 frases/160 palavras):

A Islândia ocupa o topo do ranking de igualdade de gênero desde 2000.
As Ilhas Malvinas foram disputadas por argentinos e ingleses.
O território do Kosovo fazia parte do Império Romano.
A Suécia assumiu a presidência rotativa da União Europeia em 2009.
A Ucrânia mantém o segundo maior exército europeu desde o fim da União Soviética.
O Produto Interno Bruto brasileiro totaliza mais de 6 trilhões de dólares.
A Dinamarca tem a monarquia mais popular da Europa.
As Ilhas Faroé são um território dependente da Dinamarca.
As costas groenlandesas são banhadas pelo Oceano Atlântico.
A paisagem ucraniana é formada por estepes e planaltos.
O fim do regime comunista tchecoslovaco ocorreu em 1989.
A Eslováquia é Estado-membro da União Europeia desde 2004.
A Rússia faz a maior de suas fronteiras com a Mongólia.
Os paraguaios perderam parte de seu território após a guerra.
Os húngaros são a maior minoria étnica na Eslováquia.
Os montes Tatra ficam na cadeia montanhosa dos Cárpatos.

Geografia/História (24 frases/160 palavras):

Brasília é a capital do Brasil.
O clima do Paraná é subtropical.
A Igreja Ortodoxa Ucraniana possui 7540 paróquias.
O regime militar durou 21 anos.
O Sri Lanka tem duas capitais.
Liubliana é a capital da Eslovênia.
Mario Abdo Benítez é o presidente do Paraguai.
A Moldávia é o país mais pobre da Europa.
Os croatas migraram para várias partes do mundo.
A região foi colonizada por famílias suíças.
Portugal fica na Península Ibérica.
A Armênia declarou sua independência em 1991.
Botsuana possui 581730 quilômetros quadrados.

O Brasil possui 27 unidades federativas.
Tuvalu é o quarto menor país do mundo.
A Groenlândia é autônoma desde 1979.
Angela Merkel foi eleita pelo povo alemão.
Joinville é a maior cidade de Santa Catarina.
A língua oficial da Dinamarca é o dinamarquês.
A Dinamarca não adere ao euro.
O PIB dinamarquês é de 347 bilhões.
A maioria dos dinamarqueses é luterana.
A Islândia não possui exército.
A Polônia entrou na OTAN em 1999.

Fonte: elaborado pelo autor