

## DESENVOLVIMENTO DE UMA API PARA CHATBOTS DE VENDAS E GESTÃO DE SEGUROS

Willian Valmorbida<sup>1</sup>, Lucas Medeiros Hart<sup>2</sup>

**Resumo:** Com o crescente avanço tecnológico, as empresas possuem muitos dados sobre os clientes, mas poucas conseguem utilizar essas informações para melhorar seu atendimento, ou não possuem pessoal suficiente para cobrir um atendimento 24hs, pois há um custo muito elevado de operação. No mercado de seguros não é diferente, as seguradoras dividem a responsabilidade de atendimento e venda para as corretoras de seguro, que atuam mais próximos do seu público. Este trabalho apresenta uma alternativa para este problema, com o desenvolvimento de uma API para *chatbots* utilizando processamento de linguagem natural, visando auxiliar as corretoras na venda e atendimento de seus clientes. O levantamento de requisitos necessários para o desenvolvimento desta aplicação foi feito através de entrevistas com profissionais da área. A partir deste levantamento definiu-se um modelo de uma API em Python que utiliza bibliotecas do *Natural Language Toolkit* (NLTK) para o Processamento de Linguagem Natural (PLN), disponibilizando serviços em REST. Um protótipo foi desenvolvido para validar a proposta, integrando a API ao Facebook Messenger. Para a integração entre os sistemas de gestão e multicálculo o mesmo modelo de comunicação foi utilizado. Após o desenvolvimento de um protótipo, o mesmo foi disponibilizado para testes, utilizando uma corretora real. Durante os testes, os clientes da corretora puderam avaliar o desempenho do protótipo através de uma pesquisa qualitativa que revelou resultados satisfatórios; também foi elaborada uma pesquisa qualitativa ao gestor da corretora, que qualificou o protótipo como mediano, porém, ainda não podendo determinar o impacto real para sua corretora.

**Palavras-chave:** Interface de Programação de Aplicações (API). Chatbot. Processamento de Linguagem Natural (PLN). Corretoras de Seguros.

---

1 Mestre em Computação Aplicada – Unisinos-RS e professor do Centro de Ciências Exatas e Tecnológicas da Universidade do Vale do Taquari – UNIVATES;

2 Bacharel em Sistemas de Informação – Univates-RS.

## 1. Introdução

No ramo de seguros, companhias seguradoras buscam a cada dia investir em tecnologias para conhecer melhor seus segurados e reduzir os seus riscos, evitando fraudes ou ajustando seus preços para clientes de alto risco.

Para conseguir um atendimento que funcione 24 horas por dia, com baixo custo (se comparado a todos os encargos trabalhistas de uma equipe de *callcenter*) e que conheça seu público tão bem quanto sua base de dados permitir, muitas empresas utilizam assistentes virtuais automatizados ou *chatbots* para interagir com seus clientes. Segundo Paz, et al. (2017), a utilização de *chatbots*, ou ferramentas conversacionais automatizadas para o atendimento ao cliente, já vem sendo usado a bastante tempo. Estas ferramentas podem ou não fazer uso de inteligência artificial para interagir com seus usuários.

A fim de melhorar a comunicação dos *chatbots* e diminuir a sensação que o usuário tem de que estar conversando com uma máquina, muitas ferramentas estão utilizando técnicas de Processamento de Linguagem Natural (PLN) e de Inteligência Artificial (IA) para buscar uma conversação mais fluída e natural.

Para as corretoras de seguro, o principal desafio é conseguir gerir seus clientes de forma precisa e ágil, visto que, em caso de sinistro, os seus segurados precisam das informações da apólice rapidamente. Como também utilizar de ferramentas de multicálculo para cotar em várias companhias e retornar para seus possíveis clientes os melhores preços.

Atualmente existe no mercado uma grande diversidade de sistemas para gestão de seguros e para multicálculo, alguns possuindo formulários em site para que o próprio cliente possa executar sua cotação ou solicitar informações do seu seguro. Porém quando é necessário um atendimento mais humanizado, poucas corretoras possuem um atendimento 24 horas. Desta forma, percebe-se a necessidade de uma ferramenta que se comunique com o público de forma mais humanizada, que funcione 24 horas por dia e que possua sempre um atendimento padrão.

Visando atender esta necessidade, o presente trabalho desenvolveu uma Interface de Programação de Aplicativos (API) de comunicação que permite a construção e utilização de *chatbots* através de vários aplicativos de mensagens como Telegram, Whatsapp, Facebook Messenger e outros, utilizando de bibliotecas em Python como a Natural Language Toolkit (NLTK) que interpreta as mensagens enviadas pelos usuários e a processa adequando a resposta para manter uma conversação mais fluída e natural. Desenvolveu-se também um *chatbot* para comunicação entre os segurados ou possíveis clientes com a corretora, visando validar a API especificada.

## 2. Referencial Teórico

Buscando um melhor embasamento teórico para o presente trabalho, neste capítulo são apresentadas referências bibliográficas necessárias para compreensão do mesmo.

### 2.1 Inteligência Artificial

A inteligência artificial (IA) encontra-se presente no dia a dia da humanidade há bastante tempo, seja em nossa imaginação através do cinema, literatura ou afins, ou em muitas das coisas que usamos como *as smartTVs* e assistentes virtuais presentes em *smartphones*; sempre buscaram facilitar nossa vida e ajudar a resolver problemas (JONCO; SILVEIRA, 2015). Dentre os vários ramos de estudo da IA, encontram-se as tecnologias voltadas ao processamento de linguagem natural, a qual se aplica a *chatbots* e outras soluções inteligentes disponíveis em diversos produtos utilizados em nosso dia-a-dia.

### 2.2 Chatbot

Os *Chatbots* (Abreviação de Chatter Robots) são agentes conversacionais que simulam a comunicação em linguagem natural com os usuários do sistema para auxiliá-los com informações ou ações (MORAES et al., 2016).

Oliveira (2017) descreve os *chatbots* como, Interfaces conversacionais permitem o diálogo entre seres humanos e máquinas, sendo que estas máquinas têm como objetivo tentar substituir os humanos numa componente de transação informacional, a nível de diálogo. Estas interfaces podem ser assistentes de voz, ou chatbots. Na primeira componente, o diálogo é realizado através de voz, enquanto que na segunda é através de texto.

Os *Chatbots* são normalmente desenvolvidos para tratar de uma área de conhecimento específico através de sentenças da linguagem que se deseja abordar o assunto. Normalmente são acionados através de uma pergunta direta do usuário ou, quando programado para isso, procuram comentários sobre o assunto e os responde (HUANG et al., 2007).

### 2.3 Processamento de Linguagem Natural

Processamento de Linguagem Natural (PLN) é o processo computacional de entender a linguagem humana, convertê-la e processá-la nos níveis fonético, morfológico, sintático, semântico e pragmático. Após a estrutura ser mapeada para uma forma lógica é feita uma análise semântica e lexical extraindo as informações necessárias para o uso do sistema (GONZALEZ; LIMA, 2003).

Para Sousa (2015, p. 12) o processamento de linguagem natural tem o objetivo de permitir ao computador “compreender” e gerar sentenças na língua do usuário, para tanto, traduzindo a língua em formalismos inteligíveis por

máquina e fazendo uso dos modelos estatísticos para analisar padrões do comportamento linguístico.

Existem hoje no mercado várias ferramentas que ajudam os desenvolvedores na criação de sistemas inteligentes, sejam elas bibliotecas ou APIs que auxiliam os programas a utilizarem previsões de dados, processamentos de linguagem natural, detecção de padrões em imagens, entre outros. Como exemplo de bibliotecas temos Scikit-learn da Google ou NLTK em Python e as APIs do IBM Watson.

## 2.4 Sistema de Gestão e Multicálculo

Empresas do ramo de seguros, assim como de diversas outras áreas, são suportadas por sistemas de gestão, onde são geridos dados históricos de seus clientes e contratos. Além de um sistema de gestão, seguradoras costumam contar com um sistema de multicálculo de seguros, que permite que vários cálculos de seguro sejam feitos através do preenchimento de apenas um formulário, podendo reduzir em até 91% do tempo necessário para levantar todos os valores para seus clientes. No momento existem dois tipos de multicálculo, através de robôs e de *webservice*, o primeiro simula através de robôs o preenchimento dos dados no site das seguradoras. Já o cálculo através de *webservice* é feito utilizando APIs das companhias seguradoras com link direto, sem a necessidade de acesso ao site (DOS SANTOS, 2018).

## 3. Metodologia

O presente trabalho apresenta o desenvolvimento de uma API para *chatbot* cujo o foco é a venda e gestão de seguros fazendo uso de PLN. Para tal, buscou-se comprovar através de uma pesquisa exploratória. Segundo Prodanov e Freitas:

Quando a pesquisa se encontra na fase preliminar, tem como finalidade proporcionar mais informações sobre o assunto que vamos investigar, possibilitando sua definição e seu delineamento, isto é, facilitar a delimitação do tema da pesquisa; orientar a fixação dos objetivos e a formulação das hipóteses ou descobrir um novo tipo de enfoque para o assunto. Assume, em geral, as formas de pesquisas bibliográficas e estudos de caso (2013, p. 51-52).

Para obter um melhor embasamento científico, foi realizado uma pesquisa bibliográfica. Pesquisa sistemática que tem como objetivo levantar informações e conhecimento sobre PLN, *Chatbots*, experiência do usuário e Inteligência artificial, a fim de modelar e desenvolver a API proposta neste trabalho (KNECHTEL, 2014).

A fim de aprimorar as funcionalidades do protótipo proposto neste trabalho e melhorar a análise de requisitos da ferramenta, foi adotada uma

pesquisa experimental, utilizando de dados reais para obter uma avaliação mais próxima possível do cenário real. A pesquisa experimental é feita ao analisar um problema, montar hipóteses para a solução do mesmo e trabalhar em cima das variáveis que englobam a questão a ser resolvida (RUIZ, 2013).

Foi escolhido para este trabalho uma abordagem de pesquisa qualitativa, buscando conhecimento de especialistas para a construção da ferramenta proposta. Conforme Prodanov e Freitas (2013, p. 70), “Na abordagem qualitativa, a pesquisa tem o ambiente como fonte direta dos dados. O pesquisador mantém contato direto com o ambiente e o objeto de estudo em questão, necessitando de um trabalho mais intensivo de campo.”, e segundo Assis (2008, p. 14) “Na pesquisa qualitativa, a interpretação dos fenômenos e a atribuição de significados são fundamentais. É descritiva e não requer a utilização de métodos e técnicas estatísticas”.

Após a implementação do protótipo foi realizado um questionário aos corretores de seguros, técnicos da área e clientes para avaliar e coletar dados sobre a satisfação do uso da ferramenta. Estes responderam questões abertas e fechadas para se obter um parecer positivo ou negativo sobre o uso do *chatbot* nas suas atividades diárias de venda e gestão de seguros.

#### 4. Desenvolvimento

Este projeto possui como diferencial a criação de uma API que permite a comunicação com diversas interfaces de comunicação, podendo assim ser distribuído em aplicações como Facebook Messenger, Whatsapp, Telegram e outros aplicativos de mensagens que permitam a utilização de *chatbots*.

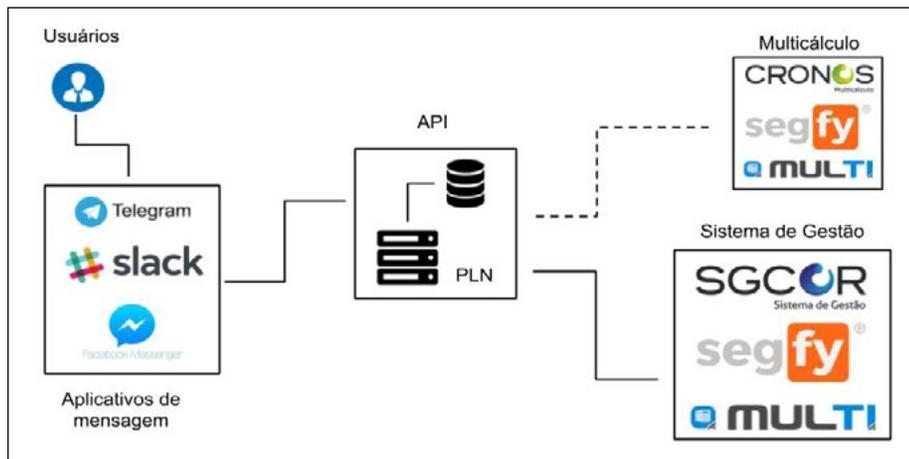
A API proposta e desenvolvida é genérica, ao ponto de se comunicar com diversos sistemas de multicálculos e gestão através de um JavaScript Object Notation (Json) permitindo uma comunicação transparente dos dados do seguro que serão enviados para a camada de multicálculo utilizando comunicação Representational State Transfer (REST), de modo a abstrair a linguagem de desenvolvimento já que trabalha em cima do Protocolo de Transferência de Hipertexto (HTTP), tendo assim uma maior compatibilidade para integração com o sistemas.

A comunicação entre os sistemas de Multicálculo são genéricas permitindo uma ampla integração de softwares do mercado a API deste projeto. A forma como os dados serão salvos nos sistemas externos ficam por conta do proprietário do mesmo, sendo necessário apenas respeitar o modelo de comunicação de saída de dados da API.

Para a comunicação com o segurado e validação da proposta, foi desenvolvido um *chatbot* integrado ao aplicativo Facebook Messenger que se comunica com a API desenvolvida, a qual é responsável por efetuar o processamento de linguagem natural, classificando qual o interesse do usuário e qual seria a melhor resposta ou questionamento a fim de conduzir a conversa.

Como ferramentas de validação para as áreas de multicálculo e gestor em seguros foram utilizados respectivamente, os softwares Cronos Multicálculo e SGCOR, conforme diagrama ilustrativo apresentado na Figura 1.

Figura 1 – Diagrama resumido da integração implementada



Fonte: Elaborado pelo autor (2018).

O Quadro 1 descreve os 4 módulos que juntos formam o sistema por completo, desde a comunicação do usuário até a busca e processamento de dados.

Quadro 1 – Módulos da API implementada

Aplicativos de mensagem	Camada de interface de comunicação com o usuário.
API	Camada responsável por salvar toda a comunicação com o usuário como também decidir para qual sistema auxiliar deverá fazer requisição ou que informação e/ou arquivo deve ser devolvido ao usuário. Camada também que irá fazer o processamento de linguagem natural e definir qual a intenção do usuário.

Aplicativos de mensagem	Camada de interface de comunicação com o usuário.
Multicálculo	Sistema responsável que irá receber os dados do item e segurado, levantados pela API através do aplicativo de mensagens que pode ou não terem sido consultados do sistema de gestão. Sendo esta camada opcional visto que caso não haja esta camada os dados podem ser enviados a camada de gestão para que seja efetuado manualmente pelo corretor.
Sistema de gestão	Sistema responsável por retornar informações sobre o seguro, como item, segurado e coberturas tanto para o usuário como também para a API para que sejam utilizados no multicálculo.

Fonte: Elaborado pelo autor (2018).

#### 4.1 Escopo

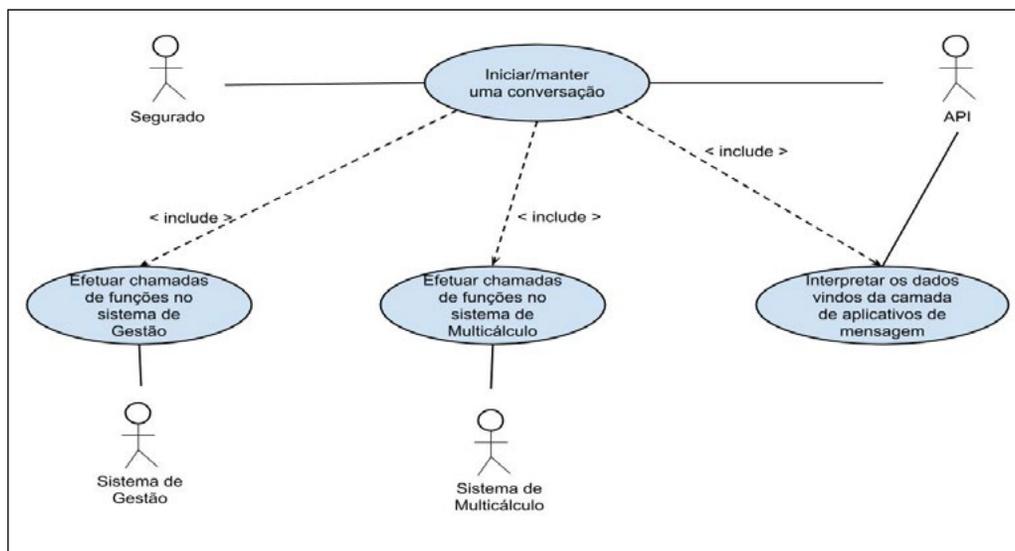
Os requisitos funcionais da implementação foram levantados através de reuniões periódicas com o gestor da ITCODE Soluções WEB, empresa proprietária dos sistemas Cronos Multicálculo e SGCOR. Softwares estes que foram utilizados para validação da comunicação entre a API desenvolvida neste trabalho e as camadas de multicálculo e gestão. Foram levados em consideração os dados e funções mais requisitados pelos segurados as corretoras.

#### 4.2 Casos de Uso

Medeiros (2010) afirma que o caso de uso, dentro da ferramenta Unified Modeling Language (UML), é o único que está presente desde o início até a finalização de um software e está diretamente ligado ao negócio. Lee e Tepfenhart (2001) descrevem o caso de uso como uma história que descreve uma sequência de eventos, solicitações e respostas entre um ator e o sistema. Para melhorar o entendimento dos requisitos da aplicação, a Figura 2 mostra o diagrama de caso de uso representando a interação dos atores com o sistema.

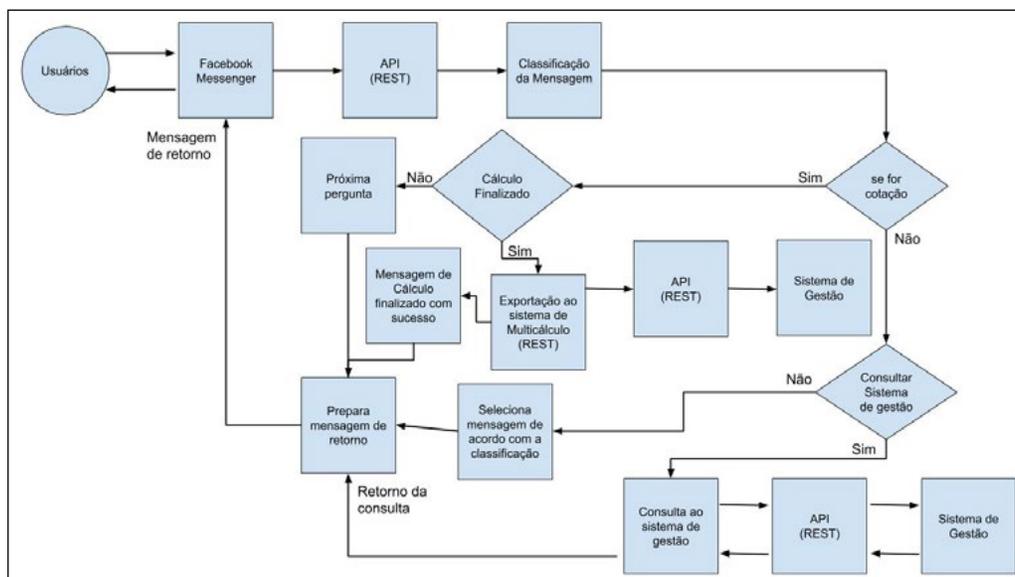
A Figura 3 descreve o fluxo das mensagens troadas entre os atores e o sistema. A mensagem enviada pelo usuário é recebida pela API que efetua a classificação da mensagem, e caso seja uma cotação de seguros o sistema envia a próxima pergunta ou a mensagem de que o cálculo foi efetuado. Caso não seja cotação o sistema avalia se há a necessidade de busca de informação no sistema de gestão ou retorna a mensagem de acordo com a classificação previamente feita.

Figura 2 – Diagrama de casos de uso



Fonte: Elaborado pelo Autor (2018).

Figura 3 – Fluxo da mensagem.



Fonte: Elaborado pelo autor (2018).

### 4.3 Modelos de comunicação

Para possibilitar a integração entre a API e os sistemas de gestão de seguros e multicálculo especificou-se um modelo de *webservices* REST que devem ser suportados por estes sistemas. Desta forma, torna-se necessário a disponibilização das Uniform Resources Locators (URLs), de cada uma das funções especificadas. Todas as informações necessárias para possibilitar a integração são mantidas no banco de dados da API.

Através da análise dos principais multicálculos do mercado, foram levantadas as informações necessárias para que o corretor possa efetuar uma cotação de seguros, sendo estas classificadas como informações do cálculo, do segurado, do item, do condutor e do questionário de risco. Estas informações foram especificadas e detalhadas com o tipo do dado, se é obrigatório e se possui alguma observação, como a dependência de um outro campo.

As informações mapeadas serviram para especificar um Json que é enviado pela API para o sistema de multicálculo, permitindo assim tal integração.

Para que a API possa ter uma conversa mais rica em informações com seus usuários é necessário que sejam feitas consultas ao sistema de gestão para buscar informações como endereço, telefone, horário de funcionamento e informações sobre a apólice de seguro.

Além destas informações, a API especificou também modelos para consulta de endereço, telefone, horário de funcionamento e apólice de seguros.

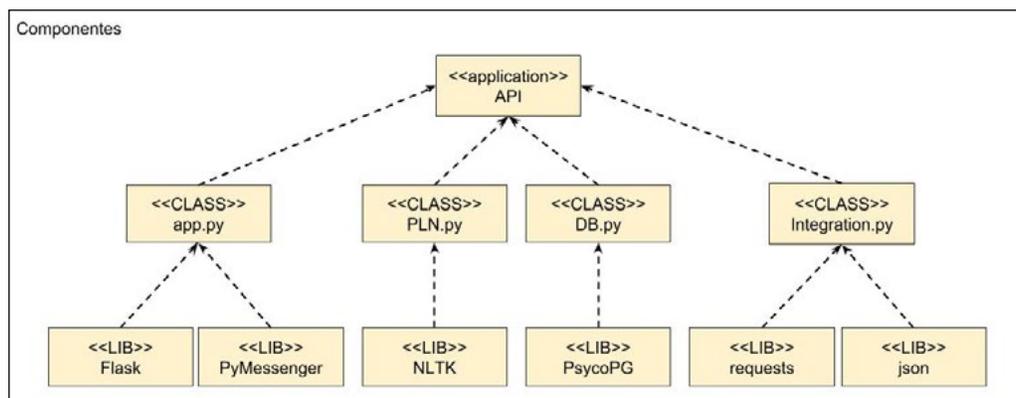
### 4.4 Desenvolvimento da API

A API possui uma camada de mensagem que se encarrega de enviar as mensagens enviadas pelo usuário, para o servidor da API, utilizando chaves de autenticação para garantir a segurança. Ao receber esta mensagem a aplicação envia as informações recebidas para uma classe específica no processamento e classificação da informação, retornando no formado adequado para a camada de mensagem, seja no formado de pergunta com opções de resposta ou texto simples.

A Figura 4 apresenta a modelagem dos principais componentes que compõe a API. A classe “app.py” (principal classe da API) utiliza as bibliotecas Flask e PyMessenger que são responsáveis pelo servidor web que recebe a comunicação externa e por enviar ao Facebook Messenger (no caso do protótipo desenvolvido) as mensagens de resposta, respectivamente. Já a “PLN.py” possui como principal dependência a biblioteca NLTK, que auxilia no processamento de linguagem natural. A classe “DB.py” utiliza a biblioteca “PsycoPG” que é responsável por fazer a conexão com o banco de dados e salvar todo o histórico das conversas entre outras informações. E por último a classe “Integration.py”, possuindo como principais dependências as bibliotecas

“json” e “requests”, é a classe responsável por fazer a integração entre a API e os sistemas de gestão e multicálculo.

Figura 4 – Dependências das principais classes do sistema.



Fonte: Elaborado pelo autor (2018).

Para medir o grau de confiança e classificar a informação enviada pelo usuário a classe “PLN.py” limpa a frase enviada, removendo todos os acentos e caracteres especiais para evitar ruídos, e compara com cada uma das classes pré treinadas e suas frases. Para isso a sentença enviada pelo usuário, limpa, é desmembrada em *tokens* (separação em palavras, removendo artigos e preposições conhecidas como *stopwords*) e cada token sofre o processo de *stemming* (processo que reduz uma palavra flexionando a sua raiz gerando o *stemm*, esta nova palavra pode ter ou não sentido. Como exemplo, para que se possa considerar palavras como amigo, amiga e amigão como a mesma palavra, retornando ‘amig’ como *stemm* da palavra) e é avaliado em relação aos treinamentos, vendo sua relevância para os treinos retornando uma pontuação.

Para calcular o grau de confiança, é dividida a pontuação da classificação encontrada com a soma de todas as pontuações das classificações, retornando um valor entre zero e um, quanto mais próximo de um mais confiante é a resposta do sistema. Caso o grau de confiança seja zero (sem classificação), o sistema retorna a mensagem “Desculpa, eu não consegui entender, poderia refazer a pergunta?” para que o usuário refaça a pergunta ou reformule-a; já quando é identificada qualquer classificação, o sistema retorna a mensagem equivalente independente do grau de confiança.

Durante o desenvolvimento deste trabalho, foram encontrados alguns problemas relacionados a identificação e categorização das palavras. Por padrão, a biblioteca NLTK utiliza como base o dicionário inglês para poder categorizar as palavras como verbo, pronomes e etc. Desta forma, para o protótipo, utilizou-se um *corpus* em português chamado de “Floresta” que atendeu as necessidades da implementação. A Figura 5 apresenta o trecho

do código que contém as funções responsáveis pelo processo de *tokenização*, *stemming* e classificação do texto. A função “*calculate\_score\_class*” é responsável por retornar o *score* de uma classe específica baseada na sentença, para isso ela percorre os *tokens* da sentença e compara através do processo de *stemming* se a palavra se encontra nas estruturas de treinamento, caso sim a pontuação geral da classe é somada com o grau de relevância do termo nos treinos, utilizando um dividido pela quantidade de vezes que essa palavra está presente nos treinos. Na classe “*clear\_sentence*”, são limpos todos os caracteres especiais, como cedilhas, acentos agudos, hifens e outros. Na classe “*classify*” são percorridas todas classificações do sistema buscando a pontuação da classe utilizando a função “*calculate\_score\_class*” separando a maior classificação para retorná-la no final do processo.

Figura 5 – código de classificação de sentenças.

```
# calcula o score das palavras
def calculate_class_score(self, sentence, class_name):
    score = 0
    # tokenize each word in our new sentence
    for word in nltk.word_tokenize(sentence):
        # check to see if the stem of the word is in any of our classes
        if self.stemmer.stem(word.lower()) in self.class_words[class_name]:
            # treat each word with relative weight
            score += (1 / self.corpus_words[self.stemmer.stem(word.lower())])
    return score

# classifica e retorna a classe da sentença
def classify(self, sentence):
    self.iterations += 1
    high_class = None
    high_score = 0
    total_score = 0

    for c in self.class_words.keys():
        score = self.calculate_class_score(self.clear_sentence(sentence), c)
        total_score += score
        if score > high_score:
            high_class = c
            high_score = score

    try:
        confidence = high_score / total_score
    except:
        confidence = 0
    return high_class, confidence

def clear_sentence(self, sentence):
    return normalize('NFKD', sentence).encode('ASCII', 'ignore').decode('ASCII')
```

Fonte: Elaborado pelo autor (2018).

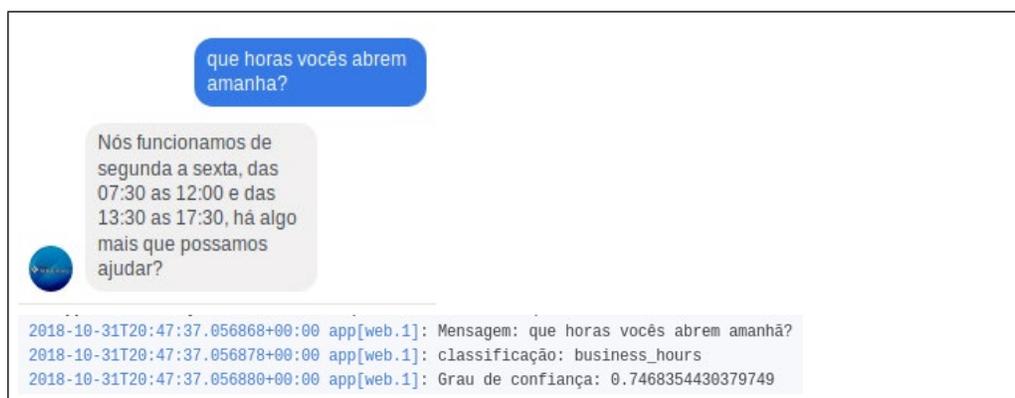
Para o treinamento do sistema foi criada a classe “*trainingHelper.py*”, que possui como única função prover as classificações e suas frases de treinamento, dessa forma foi possível treinar a API para responder adequadamente as perguntas.

Após receber a mensagem, a API verifica se é o primeiro contato do usuário nas últimas 24 horas (tempo considerado para esta API para uma nova

sessão de conversação), nesse caso é retornado um cumprimento completo, informando o nome da assistente virtual, caso haja uma cotação em aberto, o sistema encaminha para a classe responsável por avaliar em qual pergunta o usuário se encontra e qual resposta deve dar ao usuário.

A Figura 6 apresenta um exemplo de resposta ao usuário, que ocorreu ao enviar a frase “que horas vocês abrem amanhã?” ao *chatbot*, que classificou a frase do usuário como uma pergunta sobre o horário de funcionamento (*business\_hours*) e obteve grau de confiança 0.746 (equivalente a 74,6%), conforme *log* apresentado ao lado direito da imagem.

Figura 6 – Mensagem “que horas vocês abrem amanhã?” - Interface do Facebook Messenger



Fonte: Elaborado pelo autor (2018).

#### 4.5 Integração com Facebook Messenger

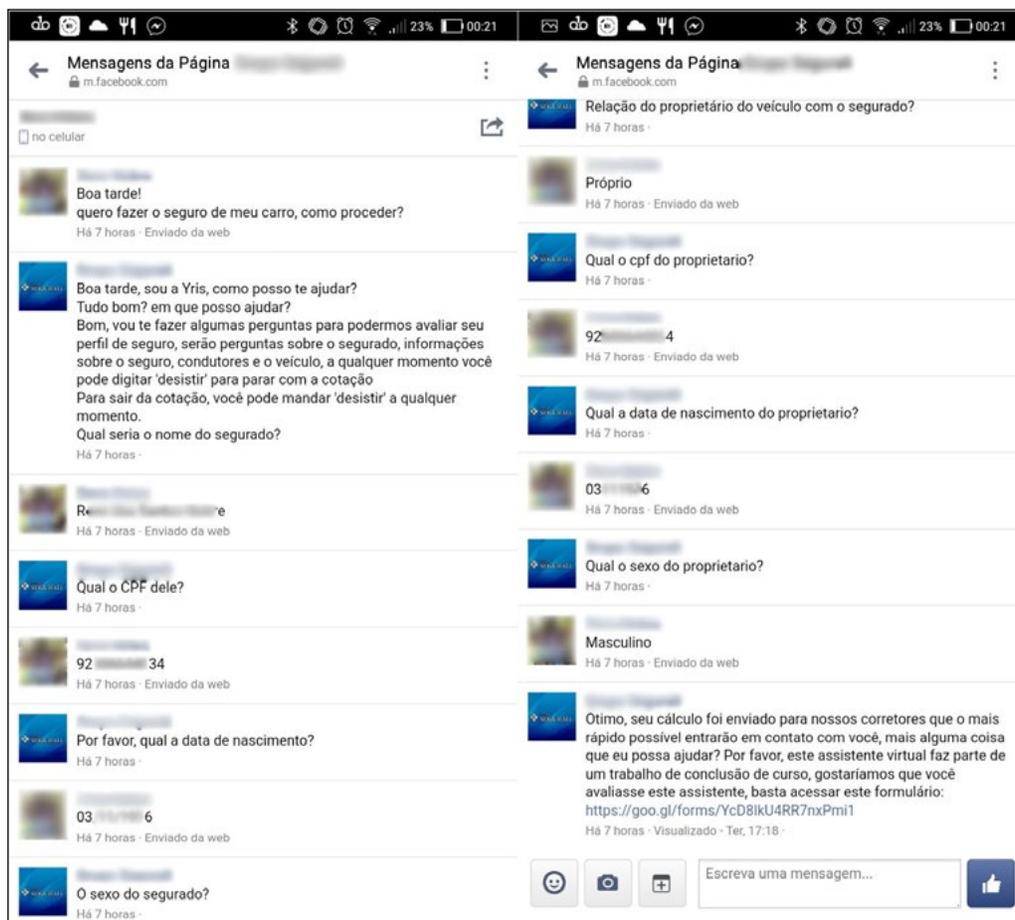
O Facebook possui compatibilidade para integração com aplicações externas, seja para logins sociais, aplicações dentro do próprio Facebook ou dentro do seu aplicativo de mensagens, o Messenger. A comunicação entre o aplicativo e a API é feita através de um *token*, que deve ser gerado pelo administrador da página. Este *token* serve para validar a comunicação entre a API e o Facebook.

Para o protótipo desenvolvido foi criado um novo aplicativo através das ferramentas para desenvolvedores do Facebook. O aplicativo criado comunica-se com a API e depois retorna para o usuário o que foi enviado pela mesma, porém o aplicativo do Facebook necessita responder uma página, por isso foi criado também uma nova página no próprio Facebook chamada *chatbot-teste* para que fosse usada como meio de interligar o Messenger e a API.

Após desenvolvimento e realização de ampla quantidade de testes, publicou-se o aplicativo, de modo a tornar possível a validação com usuários

reais. A Figura 7, mostra a conversação de um usuário real com a API através do Messenger, mostrando o início e o final da cotação (as informações pessoais do usuário foram omitidas).

Figura 7 – Conversação de um usuário real com a API.



Fonte: Elaborado pelo Autor (2018).

## 5. Validação e resultados

Para a avaliação da ferramenta, foi proposto um ambiente de teste próximo do cenário real, disponibilizando a aplicação a uma corretora. Foram submetidas duas avaliações diferentes, uma aos usuários que, através do Facebook Messenger entraram em contato com a corretora e a outra ao gestor da mesma, ambos visando avaliar qual o impacto da ferramenta proposta e desenvolvida, nas atividades realizadas por estes grupos de pessoas.

A corretora utilizada para validação do protótipo requisitou que não fosse identificada no presente trabalho, sendo identificada no mesmo como Corretora X.

Durante o período de 17/09/2018 a 18/11/2018, o *chatbot* foi disponibilizado no Facebook Messenger da Corretora X, de modo que seus clientes conseguiram entrar em contato com a corretora de forma automatizada e a qualquer horário do dia.

Durante o período em que o protótipo esteve publicado em ambiente de produção, visando sua avaliação, nenhum cliente da empresa se comunicou organicamente com a corretora. Desta forma, visando obter resultados, decidiu-se por convidar pessoas a interagir com o *chatbot* através do Facebook da empresa. Os convidados foram colaboradores da empresa em que o autor trabalha e terceiros, que mesmo não sendo utilizações espontâneas do protótipo desenvolvido, como previamente planejado para validação, são considerados como uma amostragem relevante, visto que também fazem uso de serviços de seguradoras.

A faixa etária dos clientes que preencheram o formulário de avaliação foi de 31,3% de 18 a 25 anos, 43,8% de 26 a 25 anos e 25% de 36 a 45 anos. Quanto ao sexo dos clientes, 43,8% são do sexo feminino e 66,3% do sexo masculino.

A avaliação por parte da corretora foi feita somente com o gestor da empresa. Sendo ele do sexo masculino com idade entre 26 e 35 anos, trabalhando a mais de 6 anos no seguimento de seguros. Foram elaborados dois questionários, onde o primeiro visa captar a visão dos segurados (clientes) e do gestor da corretora.

Para que os corretores entendessem melhor o propósito da plataforma, foi elaborada uma apresentação informal explicando o funcionamento e exemplos de conversação da API, para que os mesmos se sentissem confortáveis em deixar um *chatbot* responder seus contatos.

Após o período de testes, foi requisitado que o responsável pela Corretora X analisasse as respostas da API. A avaliação das interlocuções feitas entre a plataforma e os clientes da corretora foram feitas através da análise da troca de mensagens dentro do próprio Facebook Messenger, dessa forma foi possível avaliar o quanto a aplicação foi útil ao substituir uma pessoa no atendimento direto com os clientes. Questionário este composto por seis perguntas: duas para entender o perfil do corretor, três para medir o impacto do *chatbot* na corretora e uma questão aberta para sugestões e/ou críticas.

Para a avaliação do cliente final foi disponibilizado ao término das interações um formulário para levantar dos usuários se o *chatbot* supriu suas necessidades ao entrar em contato com a corretora, como também, qual foi sua experiência ao utilizá-lo e se ele voltaria a utilizar. Este questionário foi composto por oito perguntas: duas para entender o perfil, cinco para avaliar sua experiência com o *chatbot* e uma questão aberta para sugestões e/ou críticas.

O Gráfico 1 mostra a resposta dos clientes para a pergunta “Em comparação com contatos anteriores que você realizou com corretoras de seguro, sua experiência atual foi:”. A avaliação leva em consideração nota um como ruim e cinco como muito bom. Dessa forma 43,8% dos avaliados consideraram nota 5 (muito bom), 31,3% avaliaram com nota 4 (bom) e 25% avaliaram como neutro (nota 3).

Gráfico 1 - Avaliação dos clientes, pergunta sobre comparação com atendimentos anteriores



Fonte: Elaborado pelo autor (2018).

O Gráfico 2 mostra a resposta dos clientes para a pergunta “A assistente virtual supriu suas necessidades de informação?”. A avaliação leva em consideração notas de um a cinco onde um é não ajudou nada e cinco ajudou muito. Dessa forma 43,8% dos avaliados deram nota cinco, 31,3% deram nota quarto, 12,5 deram nota três e 12,5 deram nota dois.

Gráfico 2 – Avaliação dos clientes, sobre o quanto a API o ajudou.



Fonte: Elaborado pelo autor (2018).

O Gráfico 3 mostra a resposta dos clientes para a pergunta “Como você considera sua experiência ao conversar com a assistente virtual?”. A avaliação leva em consideração notas de um a cinco onde um é ruim e cinco muito bom. Neste quesito 37,5% responderam nota cinco (muito bom), 31,3% deram nota quatro, 25% deram nota 3 e 6,3% nota um.

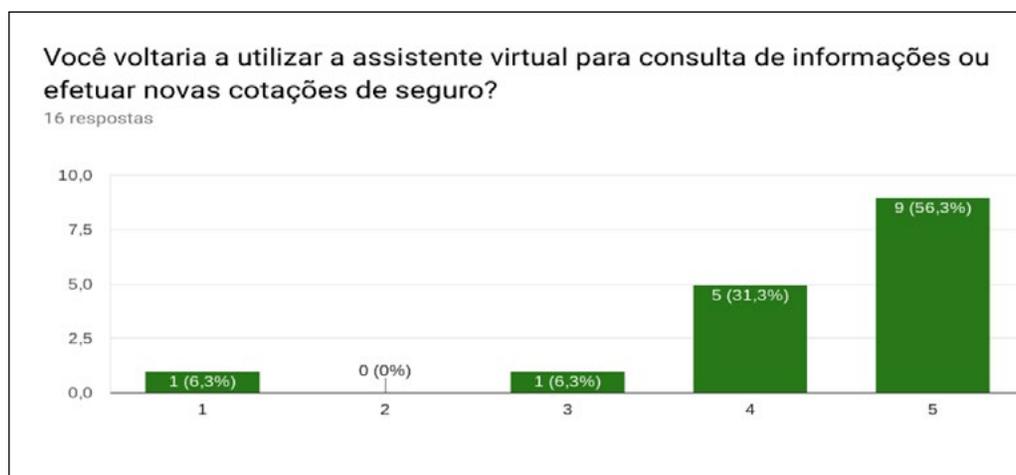
Gráfico 3 – Avaliação dos clientes, sobre a experiência ao conversar com o *chatbot*.



Fonte: Elaborado pelo autor (2018).

O Gráfico 4 mostra a resposta dos clientes para a pergunta “Você voltaria a utilizar um assistente virtual para consulta de informações ou efetuar novas cotações de seguro?”. A avaliação leva em consideração notas de um a cinco, sendo um pouco provável e cinco muito provável. Nesta pergunta 56,3% dos clientes responderam nota cinco, 31,3% deram nota quatro, 6,3% deram nota três e 6,3% deram nota um.

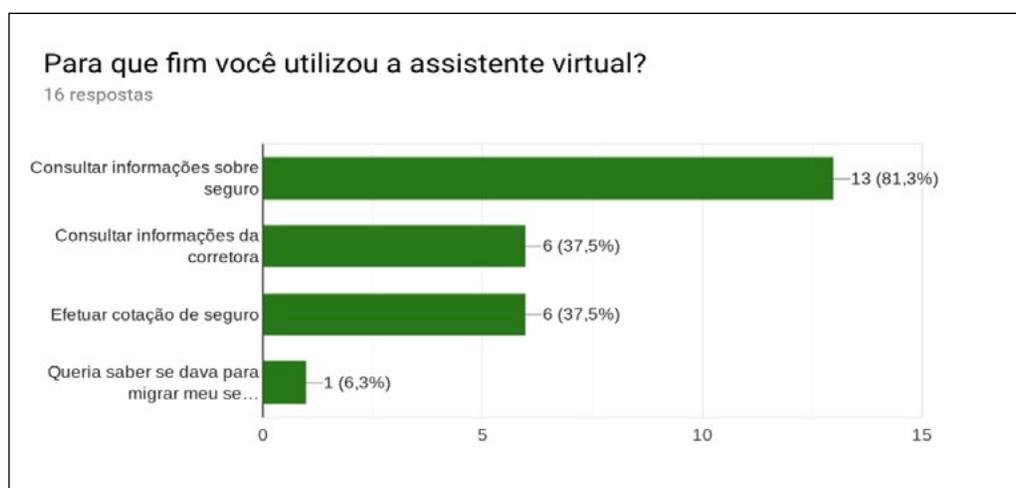
Gráfico 4 - Sobre voltar a usar o *chatbot*.



Fonte: Elaborado pelo autor (2018).

O Gráfico 5 mostra a resposta dos clientes para a pergunta “Para que fim você utilizou a assistente virtual”, sendo esta de múltipla escolha, contendo as alternativas “Consultar informações sobre seguro”, “Consultar informações da corretora”, “Efetuar cotação de seguro” e “Outros”. As respostas foram 81,3% (treze marcações), 37,5% (seis marcações), 37,5% (seis marcações) e 6,3% (uma marcação) respectivamente.

Gráfico 5 – Avaliação dos clientes, para qual finalidade ele utilizou o *chatbot*.



Fonte: Elaborado pelo autor (2018)

A última pergunta (aberta e opcional) pedia “Se você teve algum problema na conversação com a assistente virtual ou possui alguma sugestão, por favor, descreva.” apenas dois dos avaliadores responderam informando a necessidade de deixar a conversação mais fluída, ajustar erros de compreensão e reduzir o questionário de cotação de seguro.

De uma forma geral, todos os clientes que avaliaram a plataforma conseguiram utilizá-la de forma satisfatória e voltariam a utilizá-la apesar de alguns ajustes necessários para que ela ficasse mais precisa e mais completa. A comunicação provou-se funcional, conseguiu entender e responder corretamente boa parte dos questionamentos feitos pelos usuários.

O corretor responsável pela corretora estudada também foi convidado a responder algumas perguntas, sendo a primeira: “O quanto você considera útil um assistente virtual para seu negócio?”. A avaliação leva em consideração notas de um a cinco, sendo um pouco útil e cinco muito útil. Nesta pergunta o corretor respondeu neutro, com valor 3.

Para a pergunta “Como você avalia a capacidade da assistente virtual de responder as perguntas?”. A avaliação leva em consideração notas de um a cinco, sendo um muito baixa e cinco muito alta. Nesta pergunta o corretor respondeu neutro, com valor 3.

Para a pergunta “Como você avalia a assistente virtual agilizou de comunicação com seus clientes (através do Facebook)?”. A avaliação leva em consideração notas de um a cinco, sendo um atrapalhou a comunicação e cinco ajudou muito. Nesta pergunta o corretor respondeu que a aplicação não ajudou muito (resposta 2).

Para a última pergunta, aberta e opcional “Você observou algum problema na utilização da assistente virtual ou possui alguma sugestão? Por favor, descreva abaixo” o corretor informou “Ainda não há resposta para esta pergunta”.

De modo geral a avaliação do corretor foi neutra, provavelmente pelo fato de ter havido uma pequena quantidade de contatos através do Facebook Messenger durante o período de avaliação.

## 6. Conclusões

O presente trabalho teve como objetivos propor e desenvolver uma API para *chatbots* que, através de comunicação em linguagem natural auxilia-se corretoras de seguros em seu atendimento diário a seus clientes.

Após elaborar a pesquisa de bibliográfica e avaliar as soluções atuais de mercado, verificou-se que para o seguimento de seguros não existe uma opção comercial até o presente momento voltada a gestão e venda. Objetivando assim o desenvolvimento da aplicação, inicialmente através do levantamento de requisitos, modelagem, elaborado o projeto e então, o desenvolvimento da API.

Para seu desenvolvimento, foram realizados os estudos apresentados no referencial teórico, bem como o estudo de ferramentas e *chatbots* presentes no mercado, para que dessem uma maior compreensão sobre ferramentas de conversação em linguagem natural. Baseado neste estudo, assim como através do contato com profissionais da área de seguros, foi desenvolvido o modelo proposto para a aplicação, que posteriormente deu origem a um protótipo, sendo o mesmo avaliado tanto por clientes reais como pelo gestor de uma corretora.

Ao término do desenvolvimento do protótipo e realização de avaliações com clientes e um corretor, foi possível concluir que a ferramenta aqui desenvolvida possui potencial para se tornar uma solução ao que se propõe, podendo trazer ganhos de agilidade e qualidade de atendimento para as empresas, principalmente se aplicada em diversos canais de comunicação. Uma minoria das avaliações recebidas foi neutra ou negativa, e atribui-se a isso o curto período de avaliação e o fato de ser um protótipo e não um produto final, que, portanto, necessita destes *feedbacks* para ajustes e correções.

O uso de inteligência artificial e processamento de linguagem natural para a comunicação com o usuário provou-se útil, permitindo que a ferramenta respondesse aos usuários não através de comandos diretos e sim por frases, deixando assim a conversa mais fluída e natural.

Como trabalhos futuros propõe-se aprimorar consideravelmente a forma de interpretar e processar a linguagem natural para reduzir problemas de entendimento como também minimizar possíveis ambiguidades. Também há a necessidade de melhorar as classes de treino, mudança que permitiria também aumentar tanto as classificações de frases entendidas pelo sistema como deixá-lo mais preciso.

Também é necessário criar e ampliar as funcionalidades de exportação e consulta com as camadas de gestão e multicálculo para buscar informações não providas no protótipo. Além disso há a necessidade de desenvolver uma interface de gerenciamento para que os corretores possam observar todas as conversações feitas pela plataforma, como também como está o status de cada canal de comunicação.

## Referências

ASSIS, Maria Cristina de. **Metodologia do trabalho científico**. In: Evangelina Maria B. de Faria; Ana Cristina S. Aldrigue. (Org.). *Linguagens: usos e reflexões*. João Pessoa: Editora Universitária/UFPB, 2008, v. 2, p. 269-301.

DOS SANTOS, Mauricio. **O que é multicálculo de seguros auto?** Virtual, 2018. Disponível em: <<http://www.virtual.inf.br/o-que-e-multicalculo-de-seguros-auto/>>. Acesso em: 01 jun. 2018.

FERNANDES, Anita Maria da Rocha, **Inteligência artificial: Noções gerais**, Visual Books, 2008, p 1-5.

GONZALEZ, M.; LIMA, Vera L. S. **Recuperação da informação e processamento da linguagem natural**. XXIII Congresso da Sociedade Brasileira de Computação, Campinas. In: Anais do III Jornada de Minicursos de Inteligência Artificial, v. 3, p. 347-395. 2003.

HUANG, Jizhou; ZHOU, Ming; YANG, Dan. **Extracting Chatbot Knowledge from Online Discussion Forums**. In: IJCAI. 2007. p. 423-428.

JONCO, Camila M.; SILVEIRA, Stefanie C da. **Hey Siri: Inteligência artificial e a humanização dos assistentes pessoais**. Curso de especialização em cultura digital e redes sociais, Universidade do Vale do Rio dos Sinos, p. 15. 2015.

KNECHTEL, Maria do Rosário. **Metodologia da pesquisa em Educação**. Paraná, 2014

LEE, Richard C.; TEPFENHART, Willian M. **UML e C++: Guia prático de desenvolvimento orientado a objeto**, Pearson Makron Books, 2001.

MEDEIROS, Luciano Frontino de. **Banco de dados: Princípios e prática**, InterSaberes, 2013.

MORAES, Sílvia M. W.; SANTOS, André L. L.; Redecker, Matheus. **Prototipação de chatterbots como método de aprendizagem em cursos de computação: uma experiência em sala de aula**. V Congresso Brasileiro de Informática na Educação (CBIE 2016), Anais do XXII Workshop de Informática na Escola (WIE 2016), p. 91- 100, 2016.

OLIVEIRA, Sara Filipa Gomes. **Interfaces conversacionais - Chatbot para a Casa da Música**. Tese (Mestrado em Multimédia) - Universidade do Porto. Portugal, p. 154. 2017.

PAZ, Fábio J.; SILVEIRA, Clóvis; KRASSMANN, Aline; TAROUCO, Liane M.R. **Perspectivas tecnológicas para o aprimoramento de chatbots educacionais em AIML**. TE & ET, Facultad de Informática, v. único, n. 20, p. 7-15, dez. 2017.

PRODANOV, Cleber Cristiano; FREITAS, Ernani Cesar de. **Metodologia do trabalho científico: Métodos e técnicas da pesquisa e do trabalho acadêmico**. Universidade FEEVALE - Rio Grande do Sul. ed. 2, p. 277, 2013.

RUIZ, João Álvaro. **Metodologia Científica: guia para eficiência nos estudos**. 6. ed. São Paulo: Atlas, 2013

SOUSA, Aline Rocha de. **Processamento automático de línguas naturais: Um estudo sobre a localização do IBM Watson™ para o português do Brasil**. Monografia - Universidade de Brasília. Brasília, p. 76, 2015.