

Sistema integrado com Machine Learning para analisar sentimentos em pesquisas de satisfação.

LIMA, Bruno Vieira de (Autor 1);
OLIVEIRA, Poncio Elias Rodrigues (Autor 2);
MARTINGO, Lizeila Reis Abdala (Orientadora);
DUCATTI, Jose Alexandre (Coorientador)

Trabalho de conclusão de curso (Informática para negócios) – Fatec, Faculdade de Tecnologia, campus São José do Rio Preto – SP, 2022.

E-mail: bruno.lima80@fatec.sp.gov.br; poncio.oliveira@fatec.sp.gov.br;
lizeila@fatecriopreto.edu.br; jose.ducatti@fatec.sp.gov.br

Resumo: Uma das maiores dificuldades encontradas na implantação de um processo de controle de qualidade está na elaboração de um questionário simples, direto e intuitivo, que seja fácil e atraente para obter o maior número de respostas possíveis. Com o intuito de oferecer uma alternativa viável, o trabalho propõe o desenvolvimento de um sistema com a finalidade de coletar, analisar e exibir os resultados, monitorados por meio de um painel visual de maneira centralizada. Para a criação da página do formulário, são buscadas bases teóricas no entendimento dos indicadores mais assertivos sobre a satisfação de clientes e, com o apoio do *Machine Learning* na tratativa de comentários escritos pelos consumidores. Como resultado, o sistema possui mais vantagem econômica em comparação com o custo e o tempo empregado para elaborar e analisar o questionário. Dessa forma, o sistema ajuda as empresas a entenderem não só o feedback, mas o sentimento dos clientes com os indicadores traduzidos e reforçados por *Machine Learning*.

Palavras-chave: Processamento de linguagem natural. Mineração de opiniões. Análise de dados. Análise de sentimentos. Indicador consumidor. *Machine Learning*. *Feedback*.

Abstract: *One of the greatest difficulties encountered implementing a quality control process is designing a simple, direct and intuitive questionnaire that is easy and attractive to get as many responses as possible. In order to provide a viable alternative, the study proposes the development of a system with the objective of collecting, analyzing, and displaying the data in a way that it is possible to monitor the results through a visual panel, which presents the set of indicators in a centralized way. To create the the form page, theoretical bases are sought in order to understand the most assertive indicators on customer satisfaction and, with the support of Machine Learning, in dealing with comments written by consumers. As a result, the system has more economic advantage compared to the cost and time spent to design and analyze the questionnaire. Thus, the system helps companies to understand not only the feedback, but the feelings of customers with assertive indicators translated and reinforced with Machine Learning.*

Keywords: *Natural Language Processing, Opinion mining. Data analysis. Sentiment analysis. Customer indicator. Machine Learning. Feedback.*

1 Introdução

Os controles da qualidade no atendimento e do nível de satisfação dos clientes são fatores de demasiada importância na gestão estratégica de qualquer modelo de negócio. Hoje em dia, a reputação das empresas está acessível de forma muito rápida por meio de redes sociais e sites. As pessoas tendem a expressar seus sentimentos sobre produtos e serviços por meio de uma linguagem natural, em sites de reclamações, blogs, fóruns, *WhatsApp*, *Twitter*, *Facebook*, *Instagram*, páginas de vendas e sites de comentários de filmes. Analisar esses textos para extrair um conteúdo estratégico, gera um enorme trabalho e tempo por se tratar de grandes volumes de dados. Para pequenos modelos de negócios, essa dificuldade é ainda maior, pois os sistemas de análises de dados, pesquisa de marketing e criação de formulários tomam um tempo que poderia ser utilizado em novas negociações e têm um custo elevado com folhetos e mão de obra em pesquisas.

Com o avanço da tecnologia, as informações chegam cada vez mais rápidas e os consumidores estão mais exigentes, como expôs Kotler, 2000: “Os clientes de hoje são mais difíceis de agradar”, (KOTLER, 2000, p.69).

Conforme Reichheld (2003), o cliente é quem faz o marketing da empresa, colocando-se em risco ao fazer uma indicação ou sugestão. Segundo o autor, os clientes que são fiéis tornam-se boas referências, geram lucros de venda e crescimento, o que custa cinco vezes menos que o investimento em um cliente novo para a empresa. Com o crescimento das mídias sociais, os clientes conseguem expor de forma mais evidente suas opiniões, comentários e avaliações e as companhias estão expostas nas redes. Dale Carnegie vai mais além: “Quando tratarmos com pessoas, lembremo-nos sempre de que não estamos tratando com criaturas de lógica. Estamos tratando com criaturas emotivas, criaturas suscetíveis às observações norteadas pelo orgulho e pela vaidade.” (CARNEGIE, 1937, p.64). Os clientes criam suas próprias opiniões individuais de forma introspectiva e não expõem totalmente seus sentimentos sobre as empresas, produtos ou atendimento. Diante disso, é extremamente importante aprender sobre o que eles estão pensando e quais os seus sentimentos. De acordo com Yang os textos publicados são importantes fontes de informações para as empresas conseguirem estimar a opinião dos clientes (YANG et al.,2017) e saber identificar novas oportunidades de mercado passou a ser meta das áreas de tecnologia das empresas.

Uma das maiores dificuldades de quem vai implantar o processo de controle de qualidade na empresa está na elaboração de um questionário, com perguntas claras e objetivas que extraia informações relevantes para o negócio. Porque, caso o texto da pergunta seja longo demais ou não seja compreensivo, o cliente irá desistir de responder o questionário ou pode ficar confuso e não responder conforme esperado. Também, se a questão não for relevante para a gestão, ela pode gerar relatórios imprecisos, o que prejudica a gestão estratégica, pois uma vez que o questionário não seja baseado em um modelo de mensuração adequado, a qualidade das análises é diretamente afetada.

Para melhor apoiar esse questionário, é necessário também entender o sentimento dos comentários descritos em cada questão, e analisar essas opiniões por meio de um algoritmo de *Machine Learning* a fim de classificar se um comentário é positivo ou negativo. De acordo com as definições de Liu (2012), a análise dos sentimentos tem sido investigada em três níveis sendo eles:

Nível de sentença: As sentenças são avaliadas, individualmente e classificadas como sentimentos positivos, negativos ou neutros.

Nível de documento: Identifica e classifica as opiniões como um todo em dois resultados apenas sendo sentimento positivo ou negativo.

Nível de aspectos ou entidades: Analisa e identifica o sentimento das opiniões sobre uma entidade em particular, sendo esse sentimento explícito ou não na avaliação, classificando em níveis como “positivo” ou “muito positivo”.

Visando essa problemática, este projeto tem como finalidade desenvolver um sistema que aplica o questionário para a coleta de dados, analisar esses dados e apresentar em painel de visualização os resultados obtidos, baseados em indicadores da área estratégica. Para análise de textos e tratamentos dos dados não estruturados, optou-se pela classificação em nível de documento de forma supervisionada, na qual cada texto a ser analisado possui um rótulo classificatório polarizado entre negativo ou positivo, sendo possível a classificação automática das opiniões respondidas no formulário.

1.1 Justificativa

Com base nos problemas de elaboração de questionário e a necessidade de fazer uma análise assertiva do sentimento dos clientes sobre produtos, serviços e pessoas, este trabalho reunirá formas práticas de colher o sentimento dos clientes. Para este problema, poderia ser elaborado e aplicado um questionário com indicadores estratégicos, perguntas objetivas e textos curtos. É possível melhorar a experiência do usuário, criando formulários responsivos e interativos. Com a resposta do questionário aplicado, é possível unificar e consolidar as informações utilizando as tecnologias de análise de dados, mineração de texto e aprendizagem de máquina para classificar o sentimento do cliente. Posteriormente o usuário poderá consolidar as informações em formas de gráficos em um painel de visualização.

1.2 Objetivo(s)

O objetivo deste trabalho consiste em criar um questionário intuitivo e estimar por meio de indicadores a satisfação do cliente.

Desenvolver um sistema para apoiar na coleta e no armazenamento dos dados e consolidar os dados em *dashboards*, com as linguagens de programação disponíveis gratuitamente na internet.

Treinar e integrar no sistema um algoritmo de *Machine Learning* para classificar o sentimento nos comentários descritos em linguagem natural pelos clientes.

2 Trabalhos similares

A ferramenta Movidesk, tem um propósito similar com o que será desenvolvido neste projeto. Além de apoiar na aplicação do questionário e análise de dados, essa ferramenta também conta com recursos de integração, com ferramentas sociais e adequações que permitem que seja utilizada em outras áreas de negócio da empresa. Link: <https://www.movidesk.com/>.

O estudo sobre o *Net Promoter Score* (NPS) teve início em 2003, quando Fred Reicheld publicou o artigo “*One Number You Need to Grow*” na Harvard Business, o autor deu continuidade nos estudos sobre o tema, desenvolvendo ainda mais a teoria e lançou em 2006 o livro “*The Ultimate Question*” que ganhou uma nova versão em 2011. Nesse livro, Reicheld fala sobre a metodologia e como aplicá-la a cada tipo de negócio. Também, o livro explica como realizar a análise desses dados e aplicar as mudanças necessárias para o negócio, após o resultado das pesquisas.

3 Fundamentação Teórica

3.1 Indicadores

3.1.1 NPS

Para apoiar a elaboração do questionário, foram buscadas informações sobre os principais indicadores utilizadas para medir a satisfação do cliente. Para Fred Reichheld (2003), a pesquisa de satisfação do cliente pode ser baseada em um cálculo aplicado nas respostas obtidas por meio de uma única pergunta: “Qual é a probabilidade de você recomendar nossa empresa?”. A pontuação para essa resposta pode ser enquadrada em uma escala de 0 a 10, na qual é possível classificar os clientes que pontuam entre 9 e 10 como promotores. Esses são propensos a agregar comportamentos que propiciam valor ao modelo de negócio, como compartilhar sua experiência, permanecer fiel e fazer referências positivas ao produto.

Por outro lado, aqueles clientes que respondem entre 0 e 6 são classificados como detratores, pois são propensos a propagar uma reputação negativa e não voltam a utilizar o serviço ou produto. Por fim, consumidores que respondem entre 7 e 8 podem ser considerados neutros e seus comportamentos se consolidam entre detratores e promotores. Este tipo de indicador é mais assertivo em perguntas abrangentes, avaliando a satisfação no geral.

Essa métrica é denominada *Net Promoter Score* (NPS) e o seu resultado é obtido por meio da divisão da subtração de clientes promotores e detratores pelo total de clientes:

$$\text{NPS} = (\text{Clientes Promotores} - \text{Clientes Detratores}) / \text{Total de clientes}$$

A nota obtida pode ser classificada como boa ou não usando um padrão estabelecido pela própria empresa ou utilizando uma escala padrão, sendo ela:

Zona de Excelência: entre 75% e 100%;

Zona de Qualidade: entre 50% e 74%;

Zona de Aperfeiçoamento: entre 0% e 49%;

Zona Crítica: entre -100% e -1%.

3.1.2 CES

Em 2010, foi introduzida uma nova métrica chamada *Customer Effort Score* (CES) que se trata de um indicador preditivo, o qual tem por objetivo avaliar a experiência sem esforço para os clientes. Esse indicador foi mencionado por Dominique Leroy, CEO do

Proximus Group, em uma conferência STIMA de 2015 na Bélgica: “Nós nos esforçamos constantemente para satisfazer nossos clientes e diminuir sua pontuação de esforço” (LEROY, 2015). Logo, esse indicador apresenta pontos que necessitam de alterações que vão refletir positivamente na satisfação do cliente.

Esse indicador é aplicado em uma escala de 1 a 5, na qual o cliente deve informar o nível de esforço que dispôs para utilizar o serviço ou produto. O nível 1 indica um maior esforço (“difícil”) e o nível 5 aponta menor esforço (“fácil”). Este indicador pode ser calculado utilizando uma média ponderada, multiplicando a nota pelo número de pessoas em cada uma delas, dividido pelo total de pessoas envolvidas na pesquisa.

NP = Número de Pessoas

N = Nota

$$CES = ((N1 \times NP1) + (N2 \times NP2) + (N3 \times NP3) + (N4 \times NP4) + (N5 \times NP5)) / NP \text{ Total}$$

3.1.3 CSAT

O *Customer Satisfaction Score* (CSAT) é um indicador de satisfação de uso comum. Ele utiliza uma pergunta com uma escala numérica de pontuação, de 1 a 5, a qual indica o desempenho da qualidade dos serviços e produtos de qualquer tipo de seguimento. O indicador CSAT segue uma escala percentual sendo 100% o melhor índice e 0% é o pior resultado. Esse indicador, diferente do NPS, consegue medir a atual satisfação do cliente com o serviço e produto pontualmente.

Este indicador é calculado somando a quantidade de respostas entre 4 e 5 e dividindo pela quantidade total de respostas obtidas, formando deste modo um percentual de clientes satisfeitos.

$$CSAT = ((\text{Pessoas nota 4} + \text{Pessoas Nota 5}) / \text{Total de Pessoas}) * 100$$



Esse indicador pode ser utilizado no balcão de atendimento ao cliente, medindo ali a satisfação do cliente no atendimento dos funcionários, por isso é muito comum em farmácias.

É utilizado também o *feedback rating* como alternativa simplificada do CSAT, porém representado por estrelas. Esse indicador traz a resposta de acordo com a experiência do cliente, de forma simples e rápida.

3.2 Sistema

Soluções computacionais disponíveis

Para chegar às funcionalidades necessárias no sistema, foram aplicadas as tecnologias *Hypertext Markup Language 5* (HTML5) e *Javascript*. Já na estilização de interface, foi aplicado o *Cascading Style Sheets 3* (CSS3). Também, para efeitos de responsividade, foi

aplicado o *Framework Bootstrap*. Além disso, como linguagem de servidor, foi utilizado o *Microframework Web* (FLASK), aplicado na estrutura do *SnakeFramework*. Ademais, para treinamento do algoritmo de *Machine Learning* foi utilizada a linguagem *Python* e por fim, como sistema de gestão de banco de dados, foi utilizado o *My Structured Query Language* (MySQL).

3.2.1 Tecnologias Front-end (desenvolvimento da interface gráfica do usuário de um site):

Foram utilizadas as melhores tecnologias gratuitas disponíveis na aplicação do *front-end*.

Hypertext Markup Language 5 (HTML5): é uma linguagem de marcação V para a *World Wide Web* e é a tecnologia chave da internet. Neste projeto, foi utilizado o recurso de sessões que possibilita aplicar o comportamento responsivo com a versão 5.

Cascading Style Sheets 3 (CSS3): é a linguagem que define estilos para os elementos de um projeto *WEB*. A terceira versão é também a mais recente atualmente. Neste projeto, facilitou a criação das interfaces gráficas e a utilização do sistema pelos usuários.

Javascript (JS): é uma linguagem de *script* orientada a objetos multiplataforma. Neste projeto foi utilizada para criar funções dinâmicas que possibilitassem a comunicação de interface com a base de dados, para que as funcionalidades de fato sejam incorporadas.

Framework Bootstrap: é um *framework web* com código-fonte aberto para desenvolvimento de componentes de *interface* e *front-end* para sites e aplicações *web* usando HTML, CSS e JavaScript, baseado em modelos de design para a tipografia, o que melhora a experiência do usuário em um site amigável e responsivo.

3.2.2 Tecnologias Back-End (estrutura que possibilita a operação do sistema):

Microframework Web (FLASK): é uma linguagem de alto nível interpretada no servidor. Neste projeto, foi utilizada para a criação da API (Conjunto de rotinas e padrões que facilitam a comunicação e troca de informações entre sistemas) que é o meio de comunicação entre a interface gráfica e o banco de dados.

SnakeFramework: é um *framework* com código-fonte aberto criado para o desenvolvimento de sistemas web que utilizam o padrão Model, View, Controller (MVC) na linguagem *Python*. Uma das características proeminentes do *SnakeFramework* é sua sintaxe simples e organizada. Neste projeto, foi utilizado como principal ferramenta e base de desenvolvimento.

My Structured Query Language (MySQL): é um sistema gerenciador de banco de dados relacional de código aberto, utilizado na maioria das aplicações gratuitas para gerir bases de dados que se destaca pela simplicidade e capacidades de armazenagem. Neste projeto, foi utilizado para a criação dos bancos de dados.

Python: é uma linguagem de programação de alto nível interpretada de *script*, orientada a objetos, funcional, de tipagem dinâmica e forte. Neste projeto, foi utilizada para a criação do algoritmo de *Machine Learning* e a limpeza dos dados.

3.2.3 *Machine Learning*

Com os indicadores já definidos, é possível ainda colher informações por meio da adição da seguinte pergunta a cada indicador: “Em poucas palavras descreva qual o motivo da sua nota?”. Com isso, é possível realizar um trabalho de mineração sobre o texto redigido pelo cliente e ter uma segunda análise direcionada por um algoritmo de *machine learning*.

Conforme Feldman e Sanger (2007), as técnicas de mineração de textos ou dados evitam sobrecarga de informações e garantem agilidade na compreensão dos dados. Deste modo, podemos atribuir as informações a grandes conjuntos de textos, ao aprendizado de máquinas, tratando os dados por meio de um pré-processamento, por meio do qual é possível compreender se o cliente realmente está satisfeito ou não, com um texto limpo.

A mineração de opiniões ajuda as empresas a captarem o sentimento do cliente sobre os seus produtos, marca ou serviços. De acordo com Medhat et al. (2014) a mineração de opiniões identifica a polaridade de um texto digitado, descobrindo se ele está relacionado a algo positivo ou a algo negativo. Uma das formas de analisar essa polaridade de grandes volumes de dados seria a utilização de *Machine Learning* para ler, interpretar e extrair o sentimento do texto, entendendo se é positivo ou negativo. Conforme Silva (2016), a análise de sentimento é uma vasta área que unifica conceitos de aprendizado de máquina, mineração de dados, linguística, processamento de linguagem natural (PLN) e análise textual, e tem por objetivo analisar textos e determinar a atitude, emoção, opinião, avaliação ou sentimento sobre algum tópico ou entidade. (SILVA, 2016, pg. 3).

Diante disso, além da avaliação da nota dos indicadores, foi treinado separadamente do sistema, dois algoritmos de *Machine Learning*. O algoritmo que teve a melhor acurácia entre os dois treinados, foi posteriormente integrado ao sistema por meio de uma API, sendo possível traduzir se no texto descrito pelo cliente existe um sentimento positivo ou negativo.

Para esse treinamento, utilizaram-se as seguintes bibliotecas:

Scikit-Learn: A scikit-learn é uma biblioteca de aprendizado de máquina de código aberto para a linguagem de programação *Python*. Neste trabalho, foi utilizada no aprendizado de linguagem natural.

Pandas: é uma biblioteca *Python* para análise de dados, com código aberto e uso gratuito. É conhecida por sua alta produtividade e alto desempenho. O *Pandas* facilita as tarefas analíticas e elimina a necessidade de troca de ferramentas, além de possuir uma estrutura tabular em forma de lista e *DataFrames* (Estrutura usada para guardar conjunto de dados na forma de tabela).

Natural Language Toolkit (NLTK): é um conjunto de bibliotecas e programas usados para processamento simbólico e estatístico da linguagem natural, escritos na linguagem de programação *Python*. Neste trabalho foi utilizado na padronização e limpeza dos dados.

XGBClassifier: É um dos algoritmos mais utilizados por cientistas de dados, apresentando resultados superiores, principalmente em problemas de previsão envolvendo dados estruturados/tabulares. Neste trabalho foi utilizado no aprendizado de linguagem natural.

Logistic Regression: Regressão logística é um algoritmo que estima a probabilidade de ocorrência de um evento com base em um determinado conjunto de dados de variáveis independentes limitada entre “0” e “1”. Neste trabalho foi utilizado no aprendizado de linguagem natural.

Matplotlib: É uma biblioteca de *software* para criação de gráficos e visualizações de dados em geral. Neste trabalho foi utilizada para verificar o desenvolvimento dos dados.

4 Metodologia

Foi realizada uma pesquisa exploratória bibliográfica para formulação de um questionário qualitativo que ao ser aplicado, forneceria dados que seriam analisados para tomada de decisão, no âmbito da satisfação do cliente com indicadores objetivos e mais assertivos.

Além disso, buscou-se elucidar a importância de uma pesquisa de satisfação e a possibilidade de automatizar o processo de implantação, aplicação do questionário e análise de dados, com as tecnologias gratuitas disponíveis.

Com isso, foi possível representar, por meio de um modelo matemático, o problema de satisfação do cliente, de modo a obter uma base de dados sólida que forneça informações determinantes para tornar o negócio competitivo.

Foram desenvolvidos neste trabalho dois modelos de *Machine Learning* para a classificação dos textos a fim de identificar o sentimento das respostas escritas pelos clientes nos indicadores NPS, CES, CSAT e *Feedback*.

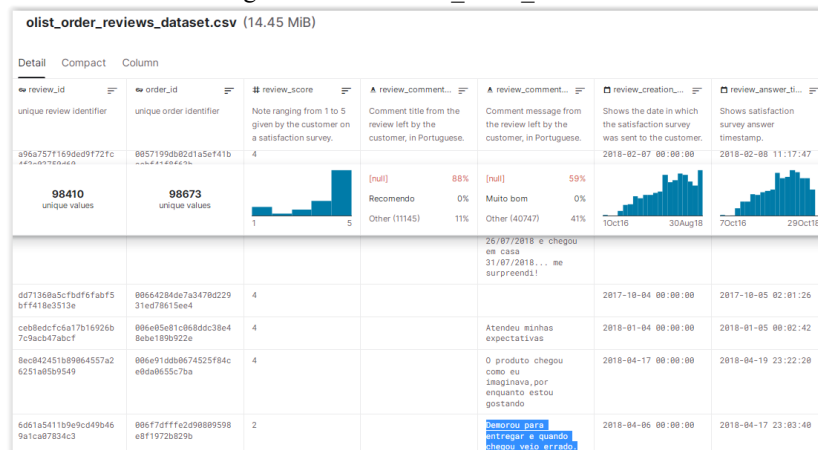
Os algoritmos verificam a polaridade de um texto digitado para identificar se é positiva ou negativa de forma supervisionada. Para isso utilizaram-se os modelos de Regressão Logística e XGBClassifier como parâmetros comparativos a fim de estimar o melhor resultado e implantar o algoritmo selecionado no sistema.

Utilizou para treino dos algoritmos a base do E-commerce extraída do kaggle (comunidade online de cientistas de dados) com avaliações e comentários de serviços e produtos e também foi utilizado a base do IMDb (plataforma de informações sobre cinema TV, música e games), com comentários e avaliações de filmes, cenários e pessoas. A finalidade é treinar um modelo de aprendizagem de máquina e linguagem natural, capaz de selecionar não só os comentários sobre produtos ou serviços, mas avaliar também ambientes, cenários e pessoas podendo assim classificar melhor os comentários nos indicadores NPS, CES, CSAT e feedback. Para isso, foram unificadas as bases para treino do algoritmo.

A base “*Olist order reviews dataset*” está disponível na comunidade kaggle com o tema “Conjunto de dados do e-commerce Brasileiro por Olist”, que possui 100.000 registros de avaliações feitas por clientes do e-commerce contendo sentimentos negativos e positivos, como mostra na Figura 1. A base do IMDb, foi extraída do próprio site contendo 49.459 comentários negativos e positivos, esta base possui dados bem diversificados de sentimentos. A unificação das bases tem por objetivo extrair o máximo de palavras possíveis para o treino do algoritmo de *Machine Learning*.

Nesse treinamento, foi utilizada a linguagem de programação *Python* (versão 3.9), com as bibliotecas descritas para *Machine Learning*.

Figura 1: Tabela olist_order_reviews



Fonte: <https://www.kaggle.com/datasets/olistbr/brazilian-ecommerce>

5 Desenvolvimento

A proposta deste trabalho é fornecer recursos que se apoiam na implantação do processo de qualidade de atendimento e seu custo de elaboração, para compará-los com as opções disponíveis no mercado. Com isso, pretende-se confirmar o baixo custo e a maior eficácia do sistema.

São abordados temas para transparecer a importância de uma pesquisa de satisfação do cliente na gestão estratégica para qualquer modelo de negócio. Também, são aplicados de maneira objetiva e clara, por meio da utilização de um modelo de questionário aprimorado e validado por especialistas da área estratégica, com uma tecnologia desenvolvida exclusivamente para este propósito.

São seguidas as etapas de processo bem definido, por meio do qual é criado um sistema *web*, com questionários focados nos melhores indicadores sobre consumidores e, para cada indicador criado, há uma pergunta escrita e um breve comentário justificando o porquê da avaliação. A justificativa descrita e armazenada e analisada por um algoritmo de *Machine Learning* capaz de identificar se o comentário referente àquele indicador interessado é positivo ou negativo. O projeto está em versão inicial e pode ser adaptado posteriormente.

5.1 Visão geral do desenvolvimento

Foi desenvolvido um sistema *web* responsivo para boa experiência de uso acessível por qualquer dispositivo, o qual pode ser integrado ao sistema corporativo da empresa. Este sistema pode ser empregado durante um contato de suporte ou enviado automaticamente por algum canal de mensagem, a fim de que o próprio cliente faça o preenchimento das informações, conforme sua experiência de uso.

As respostas de avaliações dos questionários, foram analisadas pelos critérios dos indicadores individualmente NPS, CSAT, CES ou *Feedback*.

O sistema criado disponibiliza um formulário para a aplicação do questionário, um painel de visualização dos resultados e fornece recurso em API que possibilita a integração a

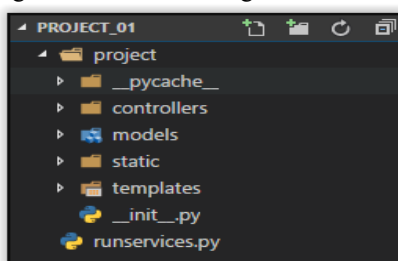
sistemas de *Help Desk*, *Call Center*, *SAP*, entre outros, que podem ser utilizados pelas empresas.

5.2 Sistema

A interface do sistema foi desenvolvida em linguagens *web* sendo elas JS, CSS e *Bootstrap* que é um *framework* de visual amigável e responsivo, muito semelhante aos aplicativos de celulares. Também, para desenvolvimento das API's, foi utilizada a linguagem FLASK, desenvolvida em linguagem *Python* e o *SnakeFramework* como base da aplicação. Além disso, para o sistema de gestão de banco de dados foi utilizado o MySQL.

O sistema foi desenvolvido em uma nova estrutura, diferente da versão atual do FLASK como mostra a Figura 2.

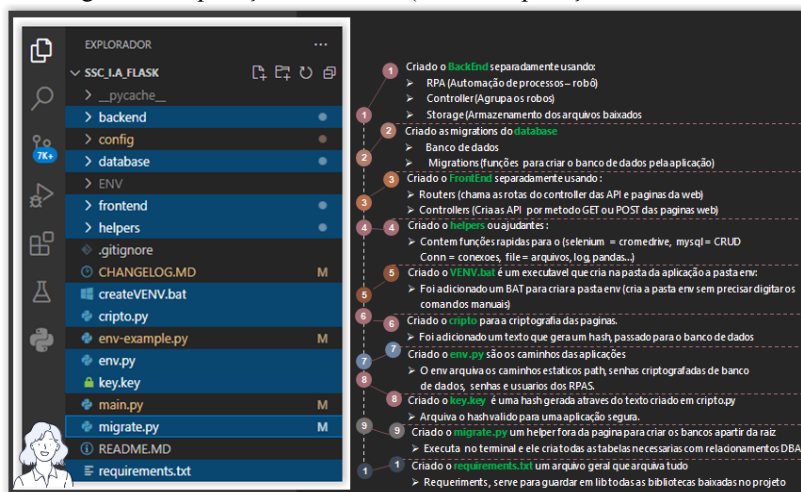
Figura 2 – Estrutura original do FLASK



Fonte: do autor

O *SnakeFramework* é oriundo das linguagens *Python* e FLASK, com a capacidade de atuação simples e objetiva entre os códigos das páginas. Segundo Douglas (2022): “Valorizar a organização e os detalhes dos códigos evita simultâneas interações de todos os lados propensas a erros”. Portanto, a melhor opção para a aplicação do sistema é o *SnakeFramework*, devido a sua organização nos códigos e nas conexões com o banco de dados, API's, segurança e interações do *front-end* e *back-end*, conforme exemplificado na Figura 3.

Figura 3 – Aplicação FullStack (Criado a aplicação no *SnakeFramework*)

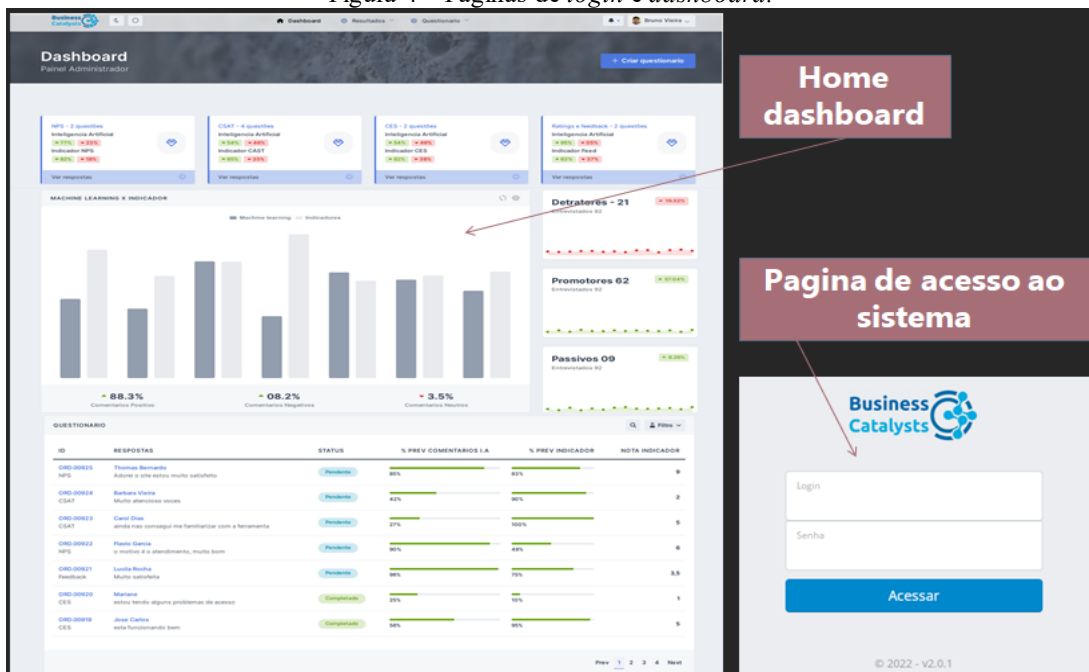


Fonte: do autor

As tecnologias foram escolhidas por se tratarem de linguagens sólidas no mercado, que têm recursos para atender às necessidades do problema a ser resolvido e pelo seu baixo custo e facilidade de implantação. Por fim, para a criação do questionário, foram utilizadas as linguagens de programação aludidas, realizadas as perguntas necessárias e o *Visual Code* foi utilizado como editor de código.

As páginas da aplicação no *front-end* possuem *login* de usuários para clientes e administradores, *dashboards* (Gráficos que ajudam no acompanhamento e exibição de indicadores) para a consolidação dos dados, perfil do usuário no sistema e a página de questionário como expostas nas figuras 4, 5 e 6.

Figura 4 – Páginas de login e dashboard.



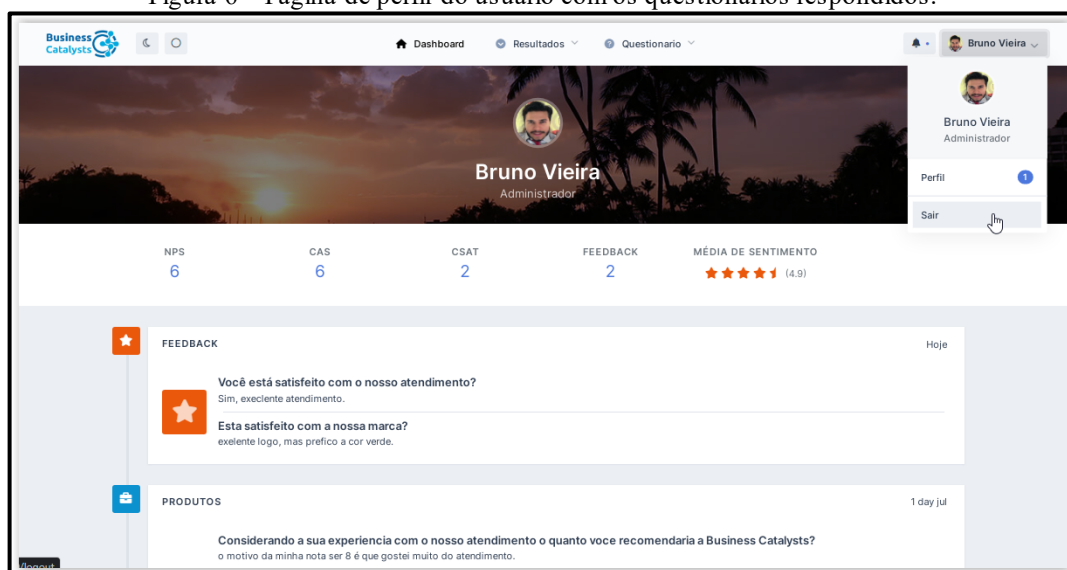
Fonte: do autor.

Figura 5 – Página e questionário do cliente.



Fonte: do autor.

Figura 6 – Página de perfil do usuário com os questionários respondidos.



Fonte: do autor.

5.3 Tratamento dos dados, treinamento do algoritmo e API *Machine Learning*

Para análise e classificação de texto, foi utilizada a linguagem *Python* com o apoio das bibliotecas: *Pandas*, as quais auxiliaram na manipulação da base de dados; *Natural Language Toolkit* (NLTK), que removeu os artigos e pronomes; *unidecode*, a qual unificou as escritas com ou sem acentuações, por meio da retirada de todas as pontuações dos textos; e *scikit-learn* que auxiliou no treinamento do algoritmo de Regressão Logística.

5.3.1 Tratamento dos dados

Para a criação e treinamento do algoritmo de *machine learning*, foi realizada uma nova limpeza dos dados com *Python* e as bibliotecas mencionadas. Para a etapa de treinamento dos modelos foi utilizada a base de comentários do IMDb no formato *Comma Separated Values* (CSV), com 49.459 comentários, dos quais 24.765 são associados ao sentimento negativo (com menos de 3 estrelas), classificados como “0” e o restante, 24.694 são associados ao sentimento positivo, (com 3 ou mais estrelas), classificados como “1”.

Da mesma forma, foi importada a base de dados do kaggle no formato (CSV) com 100.000 comentários, dos quais 15.093 são associados ao sentimento negativo (com menos de 3 estrelas), classificados também como “0” e o restante 84.907 são associados ao sentimento positivo, (com 3 ou mais estrelas), classificados como “1”. Para a limpeza dos dados iniciais achou-se necessário manter das sete colunas somente três para análise, sendo elas: “*review_comment_message*” (mensagem do cliente), “*review_score*” (nota de avaliação de 1 a 5) e “*review_comment_title*” (título do comentário, coluna de apoio).

Ao analisar as colunas título do comentário e mensagem do cliente, constatou-se que era possível extrair mensagens por meio da coluna título, uma vez que o cliente só preencheu o título e fez a avaliação, não respondendo a mensagem como exemplificado na Figura 7. Para essa problemática, foi criada uma coluna à parte, para gravar esses dados e unificar os comentários das colunas “*review_comment_title*” e “*review_score*” para nova coluna “comentário”.

Figura 7 – Análise da coluna “review_comment_title”.

review_score	review_comment_title	review_comment_message	classificacao
5	Vou recomendar sempre		NaN
5	Vendedor excelente		NaN
5	Vendedor correto.		NaN
1	Vem com defeito		NaN
5	Velo certo		NaN
5	Um loja ótima		NaN
5	tudo perfeito		NaN
5	Tudo ótimo!		NaN
3	Tudo ok...		NaN
5	Tudo OK.		NaN
5	Tudo ok!		NaN
5	Tudo ok!		NaN
4	Tudo ok		NaN
5	Tudo ok		NaN
4	Tudo OK		NaN
5	tudo Ok		NaN
4	tudo ok		NaN
5	Tudo ok		NaN
5	tudo ok		NaN
5	Tudo nos conformes		NaN

Fonte: do autor.

Outro tipo de análise e tratativa na base foi o levantamento dos dados digitados e o aproveitamento dos textos neles descritos, visto que temos dados diferentes em colunas diferentes. Para esses dados, foram concatenadas as informações, caso as colunas apresentassem diferenças de texto.

Figura 8 – Concatenação de textos diferentes.

review_score	review_comment_title	review_comment_message	classificacao	flag_diferente
4	Bom	Vou ver se o produto ,for o que prometem	1	1
4	Bom	Vou usar o produto pra ver se é o que promete ! Se não for vou de	1	1
5	Produto entregue no prazo	Vou testar o produto.	1	1
5	parabéns	vou recomendar o site , muito satisfeito	1	1
2	Troca	Vou precisar trocar o produto porque a cor da lâmpada está errad	0	1
5	otimo	vou pedir mais um kit desses	1	1
5	Otmo vendor	Vou comprar mais melhor daqui eu esperava	1	1
5	Pode confiar	Voltaria a comprar	1	1
5	EXCELENTE!	Voltarei a comprar!	1	1
5	Atenção sobre a entrega	Vocês tem que reclamar com os Correios devido a demora.Não é p	1	1
5	Amei	Vocês são 1000	1	1
1	nota 10	vocês e muito pontual	0	1
1	Péssimo.	Você vendem e quando temos problema como no meu caso manc	0	1

Fonte: do autor.

Após as tratativas dos dados, foi necessário unificar as bases Kaggle e IMDb em um único *DataFrame*, com apenas duas colunas “comentário” e “classificação”. Foram excluídos da base, todos os tipos de caracteres especiais, números e comentários vazios (NaN).

Ainda no pré-processamento realizou-se a normalização semântica das sentenças, ou seja, foram retiradas as acentuações, pontuações e *stopword* da biblioteca NLTK. Também, foram removidas as palavras sem significado semântico.

Figura 9 – Funções executadas em sequencia para tratativas dos dados e os resultados.

```
[37] #RETIRAR AS SUJEIRAS NUMEROS BARRAS INTERROGAÇÕES
      novo_df['comentario'] = novo_df['comentario'].apply(remove_stop_words_sujera)

[38] #PALAVRAS EM NUMERICOS E TEXTOS RABISCADOS E CARACTERES ESPECIAIS
      novo_df['comentario'] = novo_df['comentario'].apply(remove_stop_words_tex)

[39] #REMOVER TODAS AS ACENTUAÇÕES DAS PALAVRAS
      novo_df['comentario'] = novo_df['comentario'].apply(remove_acentos)

[40] #REMOVER TODAS AS PONTUAÇÕES DOS TEXTOS
      novo_df['comentario'] = novo_df['comentario'].apply(remove_pontuacao)

[41] #DEIXAR TODO O TEXTO COM LETRAS MINUSCULAS
      novo_df['comentario'] = novo_df['comentario'].apply(lambda frase: frase.lower())

[42] #RETIRAR ARTIGOS, PREPOSIÇÕES STOPWORDS (FOI RETIRADO OS ACENTOS DAS STOPWORDS)
      novo_df['comentario'] = novo_df['comentario'].apply(remove_stop_words_port)

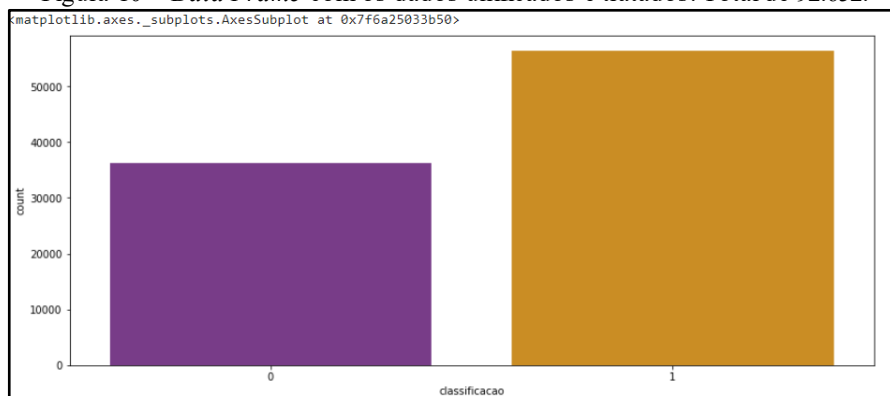
antes = df['comentario'][35490]
print(antes)
Antes da aplicação das funções
filme muito mal feito, as cenas de ação / violência são ridículas.1 ponto para a presença de Burton e Mastroianni

remove_pontuacao(df['comentario'][35490])
Após aplicação das funções
filme mal feito cenas acao violencia ridiculas ponto presenca burton mastroianni
```

Fonte: do autor

Com unificação das bases e a limpeza dos dados, o *DataFrame* apresentou um total de 92.652 comentários limpos, sendo 56.364 associados ao sentimento positivo representando 61% e 36.288 associados ao sentimento negativo representando 39%.

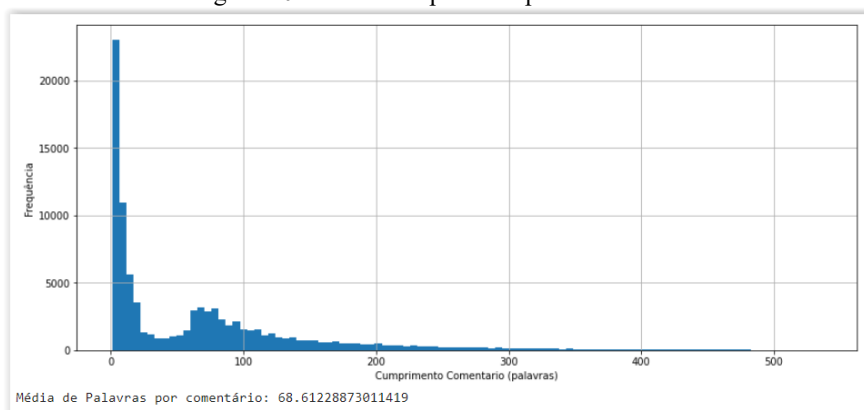
Figura 10 – *Data Frame* com os dados unificados e tratados. Total de 92.652.



Fonte: do autor

Conforme a análise de frequência, constatou-se que a média de palavras por frase é 68,61, potencializando ainda mais nosso algoritmo para prever as diversas palavras que forem digitadas no sistema, através da API. Sendo assim, com os dados tratados a base esta pronta para o treinamento do algoritmo de *machine learning*.

Figura 10 – Média de palavras por comentário.



Fonte: do autor

5.3.2 Treinamento e teste dos algoritmos

Para o treinamento foi vetorizado as palavras em colunas e as linhas em índices de comentários conforme Figura 11. Utilizou-se o *Scikit-Learn* para o aprendizado da linguagem natural, por meio de uma função que treina 80% dos dados e deixa para teste e verificação os outros 20%, ou seja, o algoritmo tem como entrada um conjunto de dados para treino e um conjunto de dados para teste. Utilizou-se a acurácia como métrica para avaliação do modelo.

Figura 11 – Matriz de palavras vetorizada

	acho	ainda	algumas	alguns	anos	antes	apenas	assistir	bem	bom	...	tao	tempo	ter	todo
0	0	0	0	0	0	0	0	0	0	0	...	0	0	0	0
1	0	0	0	0	0	0	0	0	0	0	...	0	0	0	0
2	0	0	0	0	0	0	0	0	0	0	...	0	0	0	0
3	0	0	0	0	0	0	0	0	1	0	...	0	0	0	0
4	0	0	0	2	0	0	0	0	0	2	...	0	3	0	0
...
92647	1	0	0	0	0	0	1	2	1	0	...	1	1	1	0
92648	0	2	0	1	1	2	3	0	0	2	...	3	2	3	0
92649	1	0	0	0	1	0	0	0	0	0	...	0	0	0	0
92650	1	0	0	0	0	0	0	0	0	0	...	1	1	1	0
92651	0	0	0	2	1	0	0	2	0	0	...	0	1	1	1

92652 rows x 50 columns

Fonte: do autor

A predição do algoritmo de regressão logística chega à acurácia de 87% de acerto sobre os 20% de comentários separados para o teste do modelo conforme a Figura 12. Para os comentários positivos “1”, a precisão de acerto é de 89%, e sobre os comentários negativos “0”, a precisão de acerto é de 85%.

Figura 12 – Acurácia do algoritmo de *Machine Learning*.

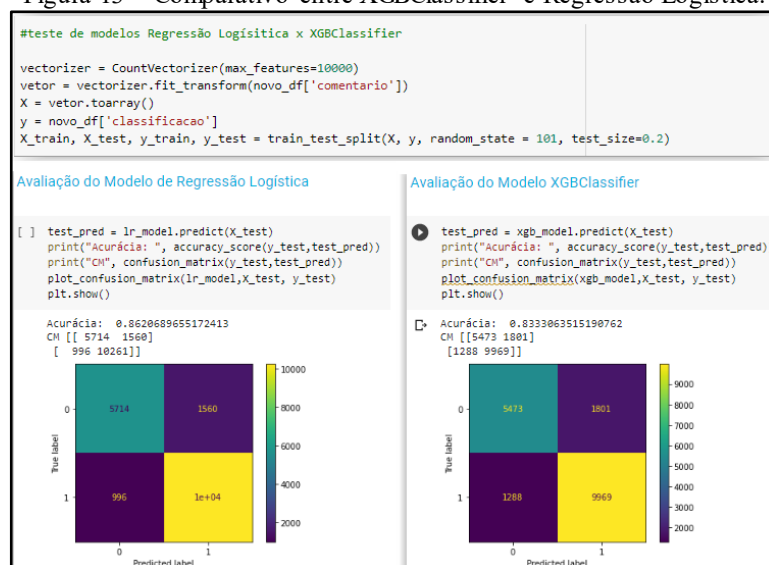
```
from sklearn.metrics import classification_report
print(classification_report(y_test, y_pred))
```

	precision	recall	f1-score	support
0	0.85	0.82	0.83	7230
1	0.89	0.90	0.89	11301
accuracy			0.87	18531
macro avg	0.87	0.86	0.86	18531
weighted avg	0.87	0.87	0.87	18531

Fonte: do autor

Foram utilizados para os treinos e comparações de acurácia dois modelos de *machine learning*, o XGBClassifier e o modelo Regressão Logística. Os resultados mostraram que o algoritmo de Regressão Logística mostrou-se superior com a acurácia de 86%, enquanto o XGBClassifier obteve uma acurácia de 83%. Para os treinos e comparações foram utilizadas 10.000 *features* (numero máximo aleatório de vocabulário que o algoritmo vai aprender).

Figura 13 – Comparativo entre XGBClassifier e Regressão Logística.

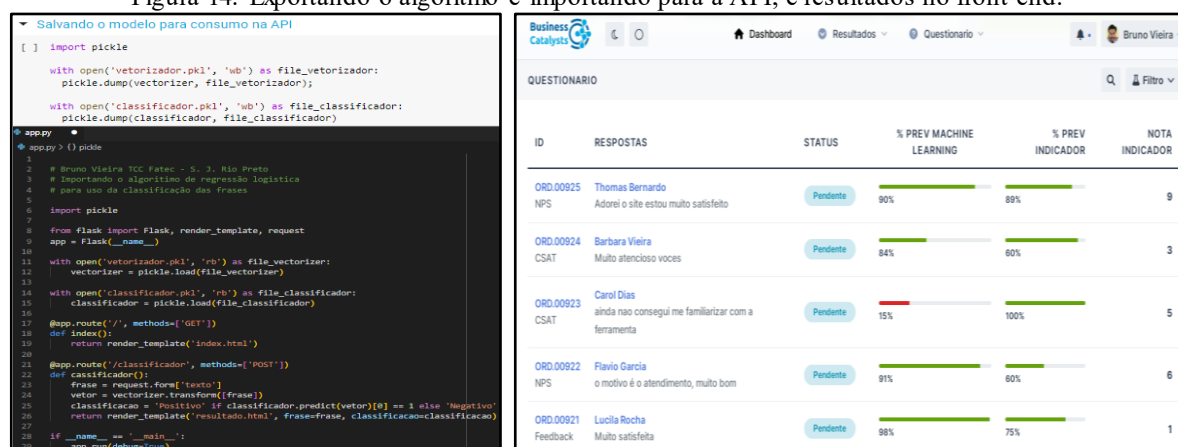


Fontes: do autor.

5.4 Aplicação do algoritmo no sistema.

A partir desses resultados e com o algoritmo definido e treinado, foi adicionado ao sistema dois arquivos *pickles* (cria protocolos binários para serializar e desserializar uma estrutura de objeto Python). Usa-se o “classificado.pkl” para armazenar o algoritmo de *machine learning* treinado e o “vetorizador.pkl” para transformar as frases em um vetor de palavras conforme mostrado na figura 11. O resultado do Front-end é acompanhado de acordo com o sentimento das frases respondidas e a nota de avaliação do indicador.

Figura 14: Exportando o algoritmo e importando para a API, e resultados no front-end.



Fonte: do autor

6 Resultados e Discussão

O modelo de *machine learning* foi bem treinado por conter uma base rica em volume de palavras com informações suficientes para elevar a preditividade do algoritmo, ou seja, as bases do E-commerce e IMDb foram utilizadas com a finalidade de treinar um modelo de aprendizagem de sentimento do cliente sobre serviços, produtos e pessoas, chegando a acurácia de 87%, com possibilidade de prescrever se o comentário deixado é positivo ou negativo.

Neste artigo, é mostrado como criar o questionário estratégico, usar uma plataforma que unifica o questionário do cliente, visualizar os dados consolidados por meio de *dashboards* e integrar no sistema um algoritmo de *machine learning* treinado. Com isso, é possível verticalizar a análise e a satisfação do cliente.

7 Conclusão

Realizar a pesquisa de satisfação por meio de questionários com indicadores objetivos pode ser um dos melhores métodos para gerir a experiência, e estimar a satisfação do cliente. Por isso, o desenvolvimento deste sistema abre um leque de oportunidades para as companhias explorarem os sentimentos dos seus clientes de forma detalhada.

Como proposto neste trabalho, o desenvolvimento da página de questionário deve ser responsivo e estruturado, para que o cliente tenha a mesma facilidade de estar respondendo a

um formulário de um aplicativo de celular, o que possibilita ao consumidor a comodidade de responder em qualquer momento ou lugar, de forma atrativa. Contar com uma ferramenta de apoio, criada com as tecnologias gratuitas disponíveis, economiza tempo, recursos e aumenta a credibilidade das análises. É possível notar como a flexibilidade do sistema favorece a criação do questionário, do painel de visualização, da coleta e do armazenamento dos dados e da integração do algoritmo de *machine learning*.

Observou-se também nesta pesquisa o treinamento de dois diferentes algoritmos de *machine learning* supervisionados e a integração no sistema do algoritmo com melhor acurácia na classificação dos comentários positivos e negativos. A classificação dos comentários feita pelo algoritmo de *machine learning* traz a possibilidade de se ter uma segunda fonte de análise assertiva no sistema, visualizando de forma verticalizada as respostas dos clientes.

Mediante o exposto, essa pesquisa oferece diferentes maneiras de analisar a real opinião do cliente em relação ao produto ou serviço oferecido, por meio de um sistema especializado, com um questionário estrategicamente elaborado. Com isso é possível colher o sentimento pela nota classificatória geral de acordo com cada indicador e analisar o sentimento do consumidor descrito nos comentários realizados junto com as notas, por intermédio de um algoritmo de *machine learning*.

8 Referências bibliográficas:

AZEVEDO, Diego. **NPS ou CSAT: quais pesquisas devo fazer dentro da jornada do cliente?**. Disponível em: < <https://www.csacademy.com.br/nps-ou-csat-quais-pesquisas-devo-fazer-dentro-da-jornada-do-cliente>>. Acesso em: 08 mai. 2022.

CARNEGIE, Dale, In: **Como fazer amigos e influenciar pessoas**. 53ª Ed. São Paulo: Companhia Editora Nacional, 2016. p. 001-336.

DUARTE, Azevedo, In: **Customer Effort Score – O que é e como medir o Indicador CES**, Disponível em: < <https://www.gp4us.com.br/customer-effort-score/>>. Acesso em: 05 mai. 2022.

DOUGLAS, Cassio.; LIMA, Bruno, In: **Porque usar o SnakeFramework?**. Disponível em: < <https://github.com/kassiodouglas/sSnakeFramework.git>>. Acesso em: 12 mai. 2022.

FELDMAN, Ronen; SANGER, James. **The Text Mining Handbook. Advanced Approaches In: Analyzing Unstructured Data**, Massachusetts, p. 410, 2007. Cambridge University Press.

KOTLER, Philip. In: **Administração de Marketing**. 10ª Ed. São Paulo: Editora Prentice Hall, 2000. p. 001-764.

LIU, B. **Sentiment Analysis and Opinion Mining**. Synthesis Lectures on Human Language Technologies, vol. 5, n. 1, p. 1-167, May 2012. Chicago: Morgan & Claypool Publishers, 2012. Disponível em: < <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.244.9480&rep=rep1&type=pdf>>. Acesso em: 20 jun 2022.

MEDHAT, W.; HASSAN, A.; KORASHY, H. Sentiment analysis algorithms and applications: a survey. In: **Shams Engineering Journal**. v.5 (4), p.1093-1113, 2014.

MELLO, Maicon. Metricas de Qualidade de Atendimento: In: **NPS e CSAT. Movidesk Artigos**, 19 Set. 2020. Seção Métricas. Disponível em: <<https://atendimento.movidesk.com/kb/article/144950/metricas-de-qualidade-de-atendimento?menuId=22478-61182-144950&ticketId=&q=>>. Acesso em 20 mai. 2022.

REICHHELD, Frederick F.; SCHEFTER, Phil. In: **E-loyalty: Your secret weapon on the Web**, Harvard Business Review, Jul. 2000. Disponível em: <<https://hbr.org/2000/07/e-loyalty-your-secret-weapon-on-the-web>>. Acesso em 09 mai. 2022.

REICHHELD, Frederick F.; SCHEFTER, Phil. In: **E-loyalty: The One Number You Need to Grow**. Harvard Business Review, Dez. 2003. Disponível em: <<https://hbr.org/2003/12/the-one-number-you-need-to-grow>>. Acesso em 20 mai. 2022.

REICHHELD, Frederick F. **The Ultimate Question: how net promoter companies thrive in a customer-driven world**. 2. ed. Boston: Harvard Business Review Press, 2011. 306 p.

SILVA, N. F. Análise de sentimentos em textos curtos provenientes de redes sociais. 2016. 141f. In: Tese (Doutorado em Ciências – Ciências de Computação e Matemática Computacional) – Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, São Carlos, 2016.

YANG, Min.; JI, Heng.; ZHAO, Zhou.; MEI, Jincheng.; ZHAO, Wei.; CHEN, Xiaojun Chen. Identifying and Tracking Sentiments and Topics from Social Media Texts during Natural Disasters. In: **Conference on Empirical Methods in Natural Language Processing**, Copenhagen, Denmark, 2017. p. 527-533.