Josué Batista Matos Deschamps de Melo
Leandro Guimarães Miranda
Marcos Vinicius Santos Souza
Vagner Ferreira Santos

Projeto Final - Análise de Reclamações de Clientes

SISTEMAS DE APOIO À DECISÃO E AO PLANEJAMENTO

São Paulo

2023

Introdução

O trabalho se trata em auxiliar uma agência governamental no âmbito financeiro como mediadora entre cliente e fornecedor. Cada cliente pode enviar uma ocorrência de determinado serviço através de um formulário online.

Com este fato torna-se complexo realizar análises para compreender cada ocorrência. Para isso surge a necessidade de utilizar modelos de NLP para classificar reclamações e realizar o encaminhamento correto.

Situação Problema

Uma agência governamental no âmbito financeiro específica, enfrenta dificuldades em lidar com a triagem manual das reclamações feitas por usuários. Atualmente, esse processo é realizado por uma equipe responsável por separar manualmente cada ocorrência, avaliar o tempo de resolução necessário para cada uma e para que possa classificá-las para o seu respectivo tipo de problema, assim podendo encaminhar as reclamações para os setores responsáveis.

No entanto, essa abordagem apresenta alguns desafios que impactam a eficiência e a qualidade do serviço prestado:

1. Separação manual de cada ocorrência: A empresa enfrenta dificuldades em separar manualmente as reclamações recebidas. O grande volume de reclamações e a diversidade de assuntos tornam o processo propenso a erros e demorado. Além disso, a falta de padronização na documentação das ocorrências dificulta a identificação e classificação correta das reclamações, prejudicando a eficácia da triagem.
2. Tempo de resolução de cada ocorrência: A equipe encarregada de realizar a triagem das reclamações, tem dificuldades em avaliar o tempo necessário para resolver cada ocorrência. A falta de informações detalhadas e a comunicação limitada entre os setores responsáveis resultam em atrasos na resolução das reclamações, o que afeta negativamente a satisfação do cliente. Além disso, a falta de um sistema de

acompanhamento adequado dificulta o monitoramento do progresso e a priorização das ocorrências com prazos mais críticos.

3. Encaminhamento de ocorrência para setor responsável: A empresa enfrenta desafios ao encaminhar as reclamações para os setores responsáveis. A falta de clareza sobre as atribuições de cada departamento e a ausência de uma abordagem padronizada dificultam a identificação do setor adequado para tratar cada ocorrência. Isso resulta em retrabalho, transferências desnecessárias e atrasos na resolução das reclamações, afetando a experiência do cliente na hora de resolver o seu problema em questão. Essa situação problemática requer uma revisão do processo de triagem das reclamações, buscando soluções que agilizem a separação das ocorrências, melhorem a precisão na avaliação do tempo de resolução e estabeleçam um fluxo claro e eficiente de encaminhamento para os setores responsáveis. A implementação de tecnologias de automação e a adoção de melhores práticas de gestão da informação podem ser consideradas para otimizar o processo de triagem e melhorar a experiência do cliente.

Com base nessas questões, nosso documento tem como objetivo propor a solução para esses desafios por meio da implementação de tecnologias de Machine Learning. A ideia é automatizar o processo de triagem das reclamações, permitindo uma eficácia superior à de um ser humano, aprimorando assim a tomada de decisão ao encaminhar cada problema ao seu respectivo setor responsável.

Solução Escolhida

Para escolher uma solução ao problema foi necessário realizar uma “triagem”, ou seja, o processo de coleta de dados feita através de uma base de dados no site Kaggle e sua devida organização. Esta organização envolve diversas técnicas de NLP.

Após estes processos iniciou-se o uso de diversos tipos de modelos de Inteligência Artificial que pudessem auxiliar nas tomadas de decisões, onde o escolhido foi o Random Forest. Apesar da escolha, foram realizados testes com outros modelos, que são: Regressão Logística, Redes Neurais, SVM e kNN.

Justificativa da solução escolhida

O Random Forest possui melhor interpretação nas análises de forma geral, justamente por se tratar de um algoritmo de classificação robusto e que é menos sensível a outliers ou ruídos nos dados. Entretanto, os outros modelos citados também possuem suas vantagens.

Regressão Logística

Para este modelo os dados foram divididos na proporção de 80% para treinamento e 20% para teste. O uso de vetorização de texto também ajudou a separar palavras nas frases obtidas dos dados.

Depois, o treinamento do modelo realiza as devidas previsões nos dados de teste. Em seguida, ocorre o cálculo da acurácia obtendo um valor de aproximadamente 70% e uma matriz de confusão para visualizar o desempenho do modelo.

Redes Neurais

Neste caso foi realizada uma classificação de texto usando uma rede neural com a biblioteca *Keras* e *Tensor Flow.*

Para se trabalhar com Redes Neurais, é necessário definir a quantidade de épocas que o modelo deve executar todo o treino. Para isso foi definido a quantidade de épocas em 10.

Feito isso, são mostradas as métricas com os seguintes valores: perda em 36%, acurácia em 87%.

SVM

O modelo *SVM* (Support Vector Machine) ajuda na classificação, mas com uma certa complexidade no código. Seus métodos são eficazes, mas podem exigir muito no custo computacional.

O código utiliza vetorização e o um treino para o *dataset* que, no final, apresenta uma acurácia de 86%.

kNN

O código realiza o treinamento e a classificação de reclamações usando o modelo Word2Vec onde basicamente o texto é transformado em um vetor de sentenças usando os vetores de palavras, facilitando o treino e exibindo palavras similares.

Em seguida é realizada a previsão usando um classificador kNN com k=5. Por fim, as métricas do modelo são calculadas, onde temos: precisão em 80%, recall em 80% e acurácia em 85%. Também é gerada a matriz de confusão.

Random Forest

Para este foram utilizados os mesmos processos de transformação de texto encontrados no modelo kNN.

É realizado o processo normal de treinamento e gerada as devidas previsões. Por último, as métricas do modelo são calculadas, onde temos: precisão em 86%, recall em 84% e acurácia em 90%. Também é gerada a matriz de confusão.

Decidindo o modelo

No final o *Random Forest* foi o vencedor, pois mesmo comparando com os outros modelos que também possuem ótimas métricas, se trata de um modelo simples que atende todas as necessidades de treino e não exige muita complexidade, como por exemplo no modelo de Redes Neurais que, naturalmente, é um modelo complexo.

Outro detalhe do *Random Forest* é referente às suas métricas, pois em comparação aos demais é o modelo que possui melhores resultados.

Refinamentos do resultado

Pré-processamento de texto

A fim de melhorar o desempenho da classificação das reclamações, exploramos diversas técnicas para o tratamento da base de dados, entre elas, *text cleaning*, *text stemming*, *text embedding* e *oversampling*.

Text cleaning é uma das primeiras etapas de otimização a serem feitas, essa técnica consiste em separar o texto em palavras (*tokenização*), remover palavras que possuem pouco significado semântico (*stopwords*) e para cada palavra restante, processamos o *stem* de cada uma delas, sendo esta última responsável por reduzir as palavras a raiz de cada uma delas, dessa forma palavras como "**marcando"** ou "**marcação"** são reduzidas a algo como "**marcar"**. Assim, conseguimos identificar as intenções e frequência das palavras mais usadas nos textos.

Então, utilizamos o *Text embedding (ou Vetorização)* para transformar nosso dado em um vetor numérico com a representação das palavras. Nesta etapa tentamos diversas abordagens usando alguns algoritmos de vetorização, entre eles, *CountVectorizer*, *HashingVectorizer*, *TfidfVectorizer* e por fim *Word2Vec*.

A técnica usada em *Word2Vec* foi a que representou o maior ganho de performance no nosso algoritmo juntamente com as outras técnicas já mencionadas.

Por fim, usamos a técnica de *oversampling* para aumentar o número de dados em classes desbalanceadas, focamos na classe "debt collection" pois a mesma apresentava muitos erros na matriz de confusão dos modelos.

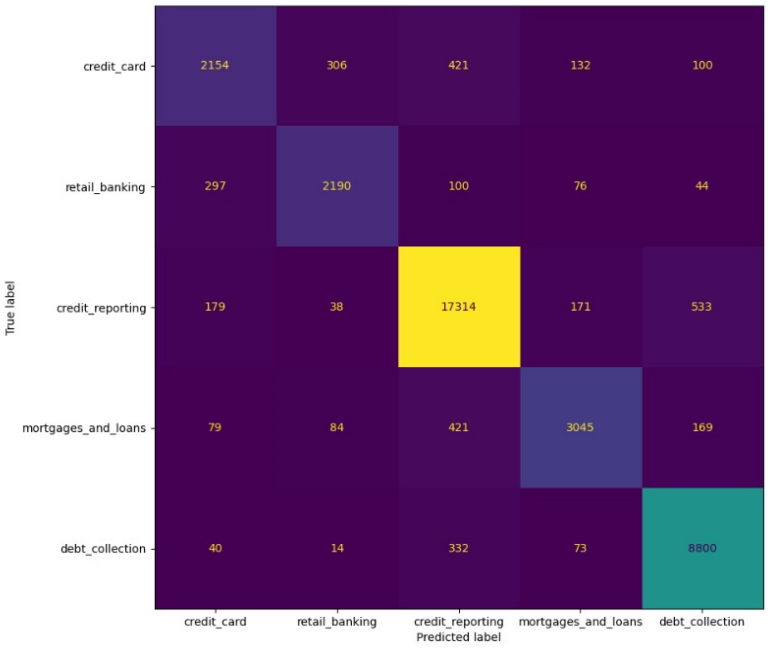
Otimizando hiperparâmetros

Após a otimização de pré-processamento de texto e escolha do modelo de *RandomForest*, fizemos alguns testes para descobrir quais hiper parâmetros trariam os melhores resultados possíveis. Esta etapa foi feita de forma manual e aqui tentamos algumas mudanças nos parâmetros de entrada do *Word2Vec*, como aumentar e diminuir o tamanho do vetor gerado, a janela de distância considerada e a contagem mínima de ocorrência de palavras. Já nos parâmetros associados a modelos testamos o número de estimadores que seriam gerados na

construção das árvores de decisão internas do modelo. Após todas as otimizações, tivemos um ganho em acurácia do modelo em torno de 10%.

Resultados obtidos

A utilização da técnica de processamento de linguagem natural *Word2Vec* em conjunto com *RandomForest* para realizar a classificação gerou um modelo com 90% de acurácia, 86% de precisão e 84% de *recall*. Realizamos o *oversampling* na base pois analisando os resultados, o modelo tendia a classificar um número relativamente alto de *debit_collection* como *credit_reporting*, cerca de 1/3 do total da classe. Avaliando o resultado da árvore de decisão a taxa de acerto foi de 94% para *credit_reporting*, 69% para *credit_card*, 80% para *retail_banking*, 80% para *mortgages_and_loans* e 95% para *debt_collection*. A superioridade para classificação do *debt_colletcion* aconteceu após o balanceamento da classe, já para *credit_reporting* acontece por causa da quantidade de dados com essa classificação na base, 56% das reclamações pertencem a esta classe.



Recomendação final

Ao final das nossas análises, chegamos a conclusão das recomendações que devem ser aplicadas para empresa aumentar sua eficiência operacional, são elas:

1. Utilizar o modelo para fazer a triagem inicial das ocorrências e reclamações para os devidos departamentos competentes.
2. Realocar equipe de forma que o setor responsável por *Credit Reporting* tenha mais colaboradores, já que os dados mostram que esse setor recebe pouco mais de 56% das ocorrências.

Com essas otimizações, esperamos que a empresa consiga aumentar a eficiência do suporte que é dado para os clientes, de forma que possam ser agilizadas as soluções para cada reclamação, gerando maior credibilidade para a empresa, pois no ponto de vista do consumidor estaria a resolver de forma rápida os problemas de os mesmo reportam, além de reduzir internamente o trabalho manual feito pelos colaboradores, permitindo que os mesmo possam trabalhar no principal produto da empresa que é resolver as ocorrências.

Principais links

Repositório do código: github.com/josueMelo/senac-sadp-analise-de-reclamacao

Avaliação do ChatGPT: chat.openai.com/share/63e6ba1e-03a5-43ba-af00-6458ba905730