Análise de Sentimento em Opiniões sobre Eventos

Joaquim A. Santos[1], **Fabrício A. Silva**[1], **Thaís R. M. Braga Silva**[1]

[1]Instituto de Ciências Exatas e Tecnológicas – Universidade Federal de Viçosa
Campus Florestal (UFV-CAF) Florestal – MG – Brasil

joaquim.sk21@gmail.com,{fabricio.asilva,thais.braga}@ufv.br

Resumo. *Aplicativos armazenam um crescente volume de mensagens postadas por seus usuários, as quais podem refletir suas opiniões acerca de determinado tema. Tal cenário torna-se o alvo do processo de Análise de Sentimento, que visa extrair a polaridade emocional expressa em determinada forma textual, a fim de se obter conhecimento relevante. Partindo desse ponto, este trabalho trata da aplicação de técnicas para processamento de linguagem natural (PLN), bem como de um método de aprendizado de máquina supervisionado, baseado no Teorema de Bayes, com o intuito de construir um modelo capaz de estimar sentimentos de opiniões de usuários do aplicativo MyMobiConf, considerando mensagens em português. Seu desempenho é avaliado por meio de métricas específicas para essa classe de algoritmos, de modo que resultados demonstram uma precisão satisfatória para o contexto em questão.*

PALAVRAS-CHAVE: Análise de Sentimento, Aprendizado Supervisionado, Processamento de Linguagem Natural

1. Introdução

Aplicativos e redes sociais se tornaram uma fonte vasta de informações, visto que envolvem um extenso e potencialmente extensível número de usuários divulgando suas opiniões sobre os mais variados temas. Como dito em [1]: ”O crescente interesse se deve à ascensão da web como arena pública para compartilhar opiniões e sentimentos acerca de todas as áreas de nossas vidas.”. Dado o contexto, a produção desse conteúdo é feita por meio de mensagens que seguem um formato padrão, proporcionando um cenário favorável para a análise de sentimento, uma subárea do processamento de linguagem natural, a qual é direcionada para a análise de emoções expressas em textos escritos por pessoas, tais como tweets, e-mails, comentários em fóruns de discussões, dentre outros [2]. Seu objetivo é construir técnicas capazes de extrair automaticamente informações subjetivas destes textos, a fim de gerar um conhecimento capaz de auxiliar na tomada de decisão [3].

Comumente, sua aplicação é feita em redes sociais, visando identificar a opinião de usuários acerca de determinado assunto, expressa em suas mensagens, como feito em [4]. Isso se torna potencialmente valioso para o ramo corporativo, dado que é possível verificar o que o público alvo de determinada empresa comenta em relação aos seus produtos e serviços. Ademais, torna-se muito interessante o emprego dessa análise em eventos com público massivo, com destaque para a área esportiva, levando a trabalhos como o de [5], que objetiva a criação de um modelo para classificação das mensagens de usuários do Twitter acerca da Copa do Mundo de 2014, realizada no Brasil.

Além dessas redes virtuais populares, aplicativos também se constituem um ótimo cenário para aplicação da análise de sentimento, já que, assim como aquelas, envolvem usuários que expõe suas opiniões por meio de mensagens. Em especial, tornou-se comum a utilização desses para tratar do registro de um conjunto de informações relacionadas a realização de eventos (i.e., congressos, feiras, simpósios, dentre outros). Tal fato possibilita um maior controle, acompanhamento e divulgação do conteúdo em questão. Como consequência, é construído um ambiente virtual no qual usuários, envolvidos nas atividades, podem interagir, expressando suas opiniões e experiências ao decorrer de sua participação nesses eventos.

Embasado no cenário descrito até então, este trabalho apresenta o desenvolvimento de um modelo para análise de sentimento baseado em aprendizado de máquina supervisionado e processamento de linguagem natural (PLN), voltado para o MyMobiConf, um aplicativo que visa o registro de dados de eventos, além de proporcionar um ambiente para interação de usuários participantes das eventualidades em questão, através de mensagens de texto. Dessa forma, o objetivo consiste em identificar a polaridade do sentimento expressa nas opiniões de usuários do MyMobiConf acerca dos eventos dos quais participaram, a classificando em positivo, negativo ou neutro [6], considerando:

- **Positivo**: Elogios e boas impressões, bem como expressões de alegria e satisfação;
- **Negativo**: Críticas construtivas ou não a algo, bem como expressões que envolvam insatisfação ou descontentamento;
- **Neutro**: Comentário objetivo que não envolva qualquer emoção, bem como mensagens simplesmente informativas.

Para tanto, foi utilizada toda a base de dados do aplicativo, coletando mensagens referentes aos eventos cadastrados e realizados desde o ano de 2016 até 2019, constituindo-se mais de 1000 mensagens de usuários, em português.

O restante deste artigo está organizado da seguinte maneira: a Seção 2 descreve o MyMobiConf e suas principais funcionalidades, a fim de se fornecer uma visão geral sobre o mesmo. A Seção 3 aborda as principais ferramentas empregadas na análise textual, bem como trabalhos que fazem aplicação da mesma. A Seção 4 descreve a metodologia utilizada para construção do modelo de classificação, detalhando, inicialmente, a coleta e rotulação dos dados a partir da base do MyMobiConf. Em seguida, são descritos os aspectos levados em consideração para o pré-processamento das mensagens coletadas, bem como o esquema para criação e treinamento do modelo. A Seção 5 trata da apresentação e discussão de resultados, abordando as métricas de desempenho para avaliação do classificador. A Seção 6 apresenta as considerações finais e intenções de trabalhos futuros.

2. O Aplicativo

O MyMobiConf é um sistema desenvolvido por estudantes de graduação do curso de Ciência da Computação da Universidade Federal de Viçosa - Campus Florestal, o qual possui uma versão de aplicação *mobile*, para Android, além de uma implementação *web*. Este é voltado para o registro e acompanhamento de eventos formais, sejam de âmbito acadêmico, profissional ou com outras finalidades específicas, visando englobar as mais diversas áreas de conhecimento, com o intuito de proporcionar maior controle de informação desses, divulgação dos mesmos, maior acesso do público ao conteúdo envolvido em tais eventos e construir um ambiente para interação virtual entre participantes e organizadores das atividades envolvidas nessas eventualidades.

Sua versão *web* é voltada para organizadores de eventos, permitindo que eles realizem o cadastro dos mesmos, bem como das atividades que os compõem: palestras, minicursos, palcos de debate, etc. Ademais, é possível informar detalhes acerca das atividades incluídas no evento, tal como horários de realização, local, responsáveis, descrição do que se tratam as mesmas, etc. Além disso, o organizador pode informar notícias relevantes sobre o evento, como mudanças nas atividades, novidades sobre seu conteúdo, possibilidade de ocorrência de sorteios e premiações, entre outras. O organizador também pode criar questionários voltados aos participantes, a fim de se ter *feedback* dos mesmos, de modo que poderá ter acesso a todas as respostas enviadas.

Outra forma de se obter o *feedbak* é por meio da abertura de um espaço para postagem de opiniões, de modo que o organizador pode visualizar o que foi dito por cada participante. Nesse espaço, é também possível a visualização das estatísticas relacionadas às opiniões, apresentadas na figura 1, além da geração de uma *Word Cloud*, exibida na figura 2, a qual ilustra as palavras mais mencionadas nos comentários de usuários (quanto maior a palavra, maior sua frequência nas mensagens).

Figura 1. Estatísticas acerca das Opiniões sobre um Evento

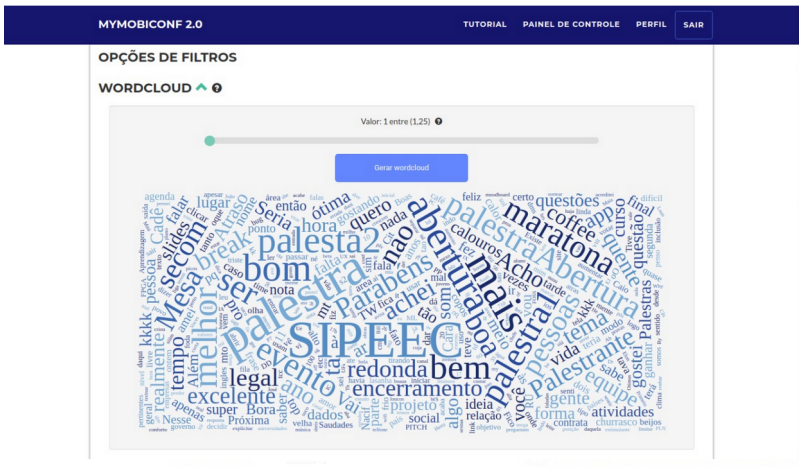

Figura 2. Word Cloud Gerada para as Mensagens de um Evento

Estatísticas também podem ser geradas para os questionários respondidos, como sobre o tempo das atividades e organização do evento, o que é ilustrado na figura 3. Adicionalmente, pode-se registrar dados dos patrocinadores dos eventos e serem visualizadas as estatísticas gerais, como avaliação dos usuários acerca do conforto térmico e sonoro do ambiente onde as atividades são realizadas, ilustradas na figura 4.

Figura 3. Estatísticas acerca dos Questionários Respondidos para um Evento

Figura 4. Estatísticas Gerais sobre o Evento

Em relação à versão *mobile*, esta é voltada para os participantes, podendo realizar um *login* (apenas pelo *e-mail,* sem necessidade de cadastro) como interessados em um evento. Feito isto, um participante tem a possibilidade de responder aos questionários registrados pelo organizador, bem como de postar qualquer número de mensagens, inclusive durante a realização do próprio evento, além de poder visualizar todas as informações relacionadas: lista de atividades e seus detalhes, patrocinadores, notícias.

Como visto, o MyMobiConf busca gerar um maior engajamento de pessoas participantes de eventos, sejam *workshops*, semanas acadêmicas, palestras de caráter profissional e diversos outros, com foco na maior interação e *feedback*, além de proporcionar o melhor acompanhamento por parte de quem organiza as eventualidades e fortalecer a divulgação.

Por fim, ambas as versões da aplicação compartilham a mesma base de dados, a qual foi utilizada nesse trabalho, mais especificamente para obtenção das mensagens referentes às opiniões dos usuários, considerando-se todos os eventos registrados. Essas mensagens seguem o formato daquelas mostardas na figura 5. Vale ressaltar que foi feito uso do sistema *web*, a fim de realizar a integração do mesmo com o modelo de aprendizado de máquina desenvolvido, como uma nova funcionalidade para a análise das opiniões.

Figura 5. Lista de Mensagens Referente às Opiniões sobre um Evento

3. Fundamentação Teórica

Os fatores mencionados na Seção 1 contribuíram para a popularização de técnicas de análise de sentimento e mineração textual, bem como para a disseminação das pesquisas relacionadas aos temas, o que se expande constantemente desde os anos 2000, como mencionado em [7]. Não obstante, há desafios e limitações impostos pela tarefa de processamento textual e sua consequente análise, visto que se tratam de dados não estruturados, difíceis de serem compreendidos por um algoritmo, como abordado em [8]. Esses dados podem advir nos mais diversos formatos, sendo necessário um tratamento adequado para seu processamento, como no caso de [9], que trata da extração de informações de artigos científicos relacionados aos efeitos da doença *Anemia Falciforme*, utilizando formatos PDF e XML.

Em virtude desse cenário, várias ferramentas tem surgido para análise e processamento de textos, tais como a popular *Weka*, além de poderosos Frameworks, como *NLTK*, *VADER* e *scikit-learn*, para a linguagem Python. Esses fornecem ampla gama de recursos para processamento textual, assim como um conjunto de algoritmos para que efetuem a análise de sentimento com alto desempenho, para a maioria dos casos. Por exemplo, em [10], é construído um modelo para classificação de textos de mídias sociais através do *VADER*, cuja performance supera modelos do estado da arte e, em alguns cenários, é até superior à precisão de seres humanos, além de possuir uma capacidade de generalização para determinados contextos, a qual ultrapassa os modelos considerados.

Apesar das ferramentas apresentadas se mostrarem eficientes e eficazes para a tarefa proposta, essas oferecem suporte reduzido para análise de textos em português, o que atinge limites como a baixa precisão e número limitado de funcionalidades para processamento textual. Em [11], tais problemas são contornados ao passo em que propõem o uso dos principais algoritmos para classificação de opiniões em 9 línguas diferentes, o que é feito através da tradução do conteúdo de texto de outros idiomas, através de uma API da Google, para o inglês, para o qual os modelos se mostram mais precisos e com maior suporte. Entretanto, atinge-se a dependência de uma ferramenta paga e com limites estipulados para quantidade de texto traduzido, além de que, após a tradução, há conteúdo que pode perder um significado essencial para a análise, dado que nem sempre a conversão de texto entre idiomas é feita com adequada precisão.

Além disso, para algoritmos baseados em aprendizado de máquina supervisionado, os quais exigem uma base de dados rotulada para treino, há uma grande dificuldade para encontrar bases grandes e adequadas o suficiente, considerando-se o idioma português.

Partindo desse ponto, para construção do modelo de classificação desenvolvido neste trabalho, foi utilizado o método de aprendizagem supervisionada chamado *Naive Bayes*, visando transpor a barreira da análise de conteúdo textual escrito em português. Essa escolha se deve ao fato da existência de outros trabalhos na literatura que fazem uso do mecanismo de análise em questão, de forma que esses apresentam resultados interessantes [12, 13]. Resultados demonstram a presença majoritária da polaridade neutra para todo conjunto de opiniões, de maneira que essa, além de outras peculiaridades dos dados textuais utilizados, mostram sua forte influência no desempenho do modelo. Ademais, foi obtida uma precisão acima de 64% para a classificação empregada, além de ser feita a integração do classificador à versão web do aplicativo, com intuito de exibir o resultado da classificação de opiniões de cada evento, quando acessado pelos usuários.

4. Metodologia

A metodologia empregada trata das técnicas utilizadas para coletar os dados do aplicativo e realizar sua rotulação, além dos procedimentos necessários para limpeza e preparação desses dados, a fim de torná-los adequados para a análise de sentimento. Por fim, é descrita a criação do modelo, considerando a escolha do algoritmo, seu treinamento e persistência do modelo em disco.

As tarefas anteriores seguem o fluxo padrão da figura 6, resumindo as etapas em:

1. Construção;
2. Avaliação;
3. Ajustes;
4. Interpretação;
5. Implantação/Persistência.

A **construção** consiste na coleta dos dados (Data Retrieval), descrita na Seção 4.1, bem como a preparação necessária desses (Data Preparation), Essa última envolve o pré-processamento sobre o conjunto de dados coletado, a fim de torná-lo adequado para análise (Data Processing), o que é detalhado na Seção 4.2, além disso, engloba a seleção dos atributos (Feature Extraction e Feature Scaling). Esses consistem nas características dos dados a serem utilizadas para a análise pelo modelo de classificação. Em seguida, deve-se escolher o algoritmo de aprendizado de máquina para formulação desse modelo, bem como definição de seus parâmetros (Modeling), tarefas que são descritas na Seção 4.3.

Finalizada a construção, deve-se efetuar o treinamento e **avaliação** do modelo, que consistem em torná-lo apto a realizar a classificação dos dados e, então, aplicar as principais métricas de desempenho sobre o mesmo, com o objetivo de verificar o quão precisa é sua rotulação. A partir dos resultados obtidos, pode-se efetuar **ajustes**, que tratam de alterações nos parâmetros do algoritmo, além de executar novos processamentos nos dados de entrada, juntamente de variações nos atributos, para que o modelo seja novamente treinado e avaliado, em busca da melhoria de seu desempenho (Model Evaluation e Tuning e ciclo representado na figura 6).

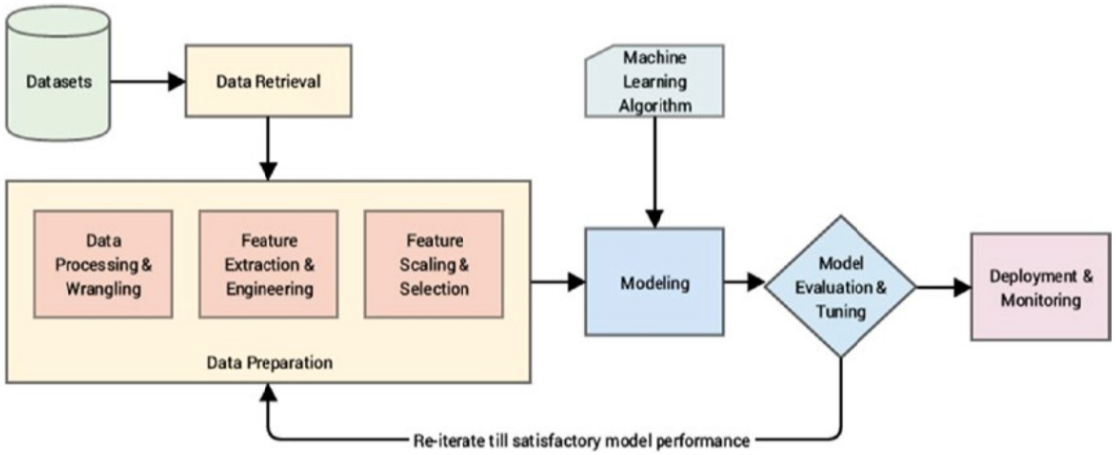

Figura 6. Fluxo Padrão para Construção de um Modelo de Aprendizado de Máquina - Fonte: Practical Machine Learning with Python (Dipanjan Sarkar, Raghav Bali e Tushar Sharma, 2018)

Para que os ajustes sejam feitos corretamente, deve-se fazer a **interpretação** dos resultados da avaliação, o que consiste em verificar do que se trata cada métrica e o significado de seus valores para o contexto em questão, com o intuito de identificar as fragilidades do algoritmo e potenciais pontos de melhoria. Por fim, ao se alcançar o modelo com desempenho satisfatório, é feita sua **implantação/persistência**, a qual se trata de definir a execução do classificador em um ambiente de produção, como um Servidor Web, para que possa ser utilizado e seu desempenho seja monitorado (Deployment e Monitoring). Ademais, é feita a persistência desse em disco, de maneira que possa ser carregado para execução sem a necessidade de sua construção novamente.

As etapas descritas nos dois últimos parágrafos são também detalhadas na Seção 4.3.

4.1. Coleta e Rotulação dos Dados

A base de dados utilizada consiste no conjunto de mensagens de usuários do aplicativo MyMobiConf, referentes às opiniões dos mesmos sobre os eventos de que participaram. Foram incluídos todos os eventos presentes na base de dados do aplicativo, considerando o período de 2016 a 2019, o que engloba: semanas acadêmicas dos cursos de graduação, como Ciência da Computação e Engenharia de Alimentos, e do nível técnico, como Técnico em Informática e Eletrônica; *workshops* de empresa júnior; palestras de empreendedorismo e etc.

A base consiste em um conjunto de quase 1000 opiniões, de caráter objetivas e subjetivas, as quais podem envolver qualquer conteúdo, mas com foco na visão dos usuários sobre sua participação nas atividades envolvidas. Para coleta, foi feita a consulta ao banco de dados relacional da aplicação. A seguir, foi dado início ao processo de rotulação das mensagens. Nessa etapa, cada mensagem deve ser rotulada com a polaridade do sentimento expressa pela mesma (Positivo, Negativo, Neutro), o que teve de ser feito manualmente, devido à ausência de um mecanismo para esta tarefa.

O que foi observado durante o procedimento descrito, é que há um certo desba-

lanceamento entre as polaridades, de modo que opiniões neutras estão em predominância, devido à maior objetividade das mensagens, além de um número muito menor de opiniões negativas. Isto influencia diretamente no treinamento do classificador e sua consequente precisão.

Ao final do procedimento, foi obtido o conjunto de todas as mensagens, referentes a todos os eventos, rotuladas de acordo com a polaridade expressa por cada uma. Além do texto, cada mensagem contém seu identificador único, bem como o de seu usuário e evento correspondente. Vale ressaltar que, embora não seja um conjunto tão extenso de dados, o mesmo foi suficiente para se treinar o classificador de modo a obter precisão considerável para o contexto tratado, além de que foi demandado maior tempo para rotulação manual. Por fim, as mensagens coletadas e etiquetadas com suas respectivas polaridades foram gravadas em arquivos CSV, onde cada linha deste arquivo corresponde a uma mensagem e seu rótulo, e posteriormente utilizadas para efetuar uma análise de sentimento baseada em probabilidade.

4.2. Pré-Processamento

Anteriormente ao seu uso, o conjunto de dados deve ser pré-processado, de modo a se fazer uma ”Limpeza dos Dados”, a fim de se remover ruídos e reparar os dados para análise, os deixando em um formato mais apropriado para construção do modelo e seu treinamento. De modo geral, se tratam de textos com conteúdo variado, semelhante a mensagens de tweets, sendo frases curtas. Nesses pode haver:

1. Números em meio ao texto;
2. Palavras concatenadas de números;
3. Links para sites Web;
4. Palavras junto dos caracteres @ e #, comumente usados para referenciar usuários e assuntos debatidos;
5. As diversas pontuações usadas na escrita;
6. Ícones de *emojis* para expressão de emoções.

Tais aspectos foram levados em conta para processar o conjunto de dados de entrada. Porém, outra questão teve de ser tratada antes: os textos, gravados em arquivo, não estavam entre aspas duplas, o que impossibilita a leitura, já que as pontuações e caracteres especiais nos textos não seriam corretamente tratadas pelos métodos de leitura de arquivos CSV. Desse modo, para tratar esse ponto, foi utilizada a ferramenta Excel, a qual permitiu padronizar os arquivos, fazendo o devido agrupamento dos textos de modo a permitir a leitura por um método Python. Então, foi gerado um novo CSV para leitura no programa.

Para a parte que trata das peculiaridades dos dados de entrada descritos, a fim de realizar uma limpeza dos mesmos, foi utilizada a biblioteca *NLTK*. Por meio dessa, foram removidos todos os tipos de links que poderiam estar nos textos, visto que esses só interfeririam na análise e são um ruído, o que foi feito por meio da substituição por expressões regulares. Essas expressões também foram utilizadas para remover as palavras concatenadas de números, já que normalmente não possuem um sentido relevante para a análise e tenderiam a equívocos pelo classificador.

As pontuações também foram removidas, já que não possuem valor para algoritmos de análise de sentimento e constituiriam apenas uma carga no processamento. Todavia, foi deixada a pontuação de ’!’, pois esta normalmente indica uma expressão forte de

sentimento em uma frase, podendo ser usada para que o algoritmo encontre um padrão em seu uso.

As *stopwords* foram removidas, já que se tratam de palavras que não agregam significado relevante à sentença, tais como preposições, advérbios e outras instâncias do idioma que não são úteis para a análise. Para tanto, usou-se a própria lista de *stopwords* do português da *NLTK* junto de outras disponibilizadas em repositórios públicos do *GitHub*.

Não foi feita a aplicação da técnica de *stemming*, "que pode ser entendido como processo de extração do radical de uma palavra" [14], pois, apesar de permitir a redução da dimensionalidade dos dados, a mesma traz desvantagens para os algoritmos de análise de sentimento baseados em aprendizado supervisionado. Tal fato ocorre porque a técnica reduz um conjunto de palavras a um mesmo radical, de modo que essas palavras podem ter sentidos bem diferentes em diferentes frases, o que irá fazer com que o algoritmo não leve essas diferenças em consideração. Portanto, é influenciado a avaliar frases com as palavras em sua forma base de maneira enganosa. Vale ressaltar que isso se aplica para qualquer que seja o idioma em que se encontram as palavras, desse modo a escolha de não aplicação do *stemming* não foi devido ao fato de as mensagens estarem em português.

Um exemplo que ilustra potenciais equívocos na análise, causados pelo uso de *stemming*, pode ser observado nas seguintes frases:

- **Sentimento Positivo:** "Adorei aquele carro!"
- **Sentimento Negativo:** 'O barulho da carruagem me incomodou muito."
- **Sentimento Positivo:** "O passeio de carro foi ótimo!"
- **Sentimento Negativo:** "A carruagem fez com que eu ficasse preso no trânsito."

Aplicando-se a extração de radical sobre as sentenças acima, anteriormente à análise, tem-se que as palavras "carro"e "carruagem"serão reduzidas à mesma raiz: **carr**. Dessa forma, um algoritmo para classificação irá considerá-las como a mesma palavra, o que fará com que essa palavra seja associada tanto à frases positivas quanto negativas. Por outro lado, se não extraído o radical, as palavras em questão serão tratadas como distintas, de maneira que "carro"seja associada apenas à frases positivas e "carruagem"somente à frases negativas. Esse ponto afeta consideravelmente a probabilidade estimada por um classificador, com base na ocorrência de palavras, para determinar rótulos das mensagens.

Por fim, os números separados de palavras também foram considerados, dado que esses podem influenciar positivamente no aprendizado do algoritmo, dependendo do modelo utilizado. Há modelos que levam em conta o relacionamento das palavras que aparecem em frases por meio de um anagrama, de maneira que possa haver um significado diferente dependendo de quais palavras estão acompanhadas uma da outra. Nesse caso, números que aparecem próximos de uma palavra podem indicar algo a mais sobre a frase (um exemplo é "Esta aula é Nota 10", onde, o '10' associado à palavra 'Nota', indica um sentimento bom). Por fim, palavras juntas do símbolo de # também foram usadas, já que podem se referir a assuntos específicos, de forma a se ter relevância no vínculo de uma polaridade a um assunto.

4.3. Criação do Modelo

Para esta etapa, foi utilizada a biblioteca *scikit-learn*, a qual oferece modelos de aprendizado supervisionado de fácil utilização e alto desempenho, além de métodos para

avaliação e testes dos modelos construídos. Fez-se uso de *MultinomialNB*, sendo este uma implementação de um modelo de aprendizado supervisionado baseado no método de análise de Naive Bayes, um dos mais populares para a análise de sentimento.

Naive Bayes é um método de aprendizagem probabilística baseado no teorema de Bayes. Segundo o mesmo, a probabilidade de uma característica estar presente ou não em uma classe (ou rótulo) não está relacionada com o fato de outra característica estar ou não presente [14].

O algoritmo de Naive Bayes apresenta resultados consideravelmente eficazes na análise de sentimento em textos. Além disso, a vantagem de seu uso para o contexto tratado é o fato de ser possível realizar o treinamento com base em um conjunto menor de dados, acrescentando que os textos são tratados como *bag of words* para serem classificados, de forma que as posições específicas das palavras nas sentenças não importam, sendo considerada a independência entre as mesmas [15], o que faz com que possa ser visto como um conjunto de itens com repetição. Essa formulação consiste na extração e seleção de atributos, os quais neste contexto consistem nas palavras que formam as mensagens.

Com base no que foi descrito, após a realização da coleta e pré-processamento necessário, bem como seleção do algoritmo, fez-se o ajuste dos hiper-parâmetros do mesmo, mais adequados ao cenário, os quais se tratam dos parâmetros cujos valores são atribuídos antes da geração do modelo. Em se tratando do *MultinomialNB*, o hiper-parâmetro que necessitou ajuste foi o *ngram_range*, o qual se refere à relação a ser construída entre as palavras de uma sentença. Para esse, o melhor valor foi uma relação 2-gran, o que significa que será levado em conta o relacionamento de palavras aos pares para cálculo de probabilidades. Esse valor foi obtido após sucessivas validações de desempenho, que retornaram melhor resultado quando usado *ngram_range* = 2, além disso, fóruns envolvendo discussões acerca do uso do algoritmo de Naive Bayes recomendavam tal valor para o hiper-parâmetro. Vale ressaltar que a decisão não foi embasada em outros trabalhos científicos pelo fato de não mencionarem essa questão.

Para treinamento foi utilizada a base de dados coletada e rotulada manualmente, como dito anteriormente, tendo-se então um conjunto de mensagens de opiniões pré-processadas, e classificadas quanto à sua polaridade: **Positivo**, **Negativo** e **Neutro**. Isso gera forte impacto no aprendizado do modelo, que depende muito da qualidade da base de dados de treino, o que é favorecido ao passo em que se amplia o conjunto de treinamento.

Para determinação dos melhores valores de hiper-parâmetros, bem como melhorias no conjunto de treino, foi feita a avaliação do modelo segundo validação cruzada para métricas importantes para classificadores, como será descrito na próxima seção. Em decorrência disso, para demais hiper-parâmetros além do *ngram_range* , os valores padrão se mostraram melhores, bem como foram feitos alguns ajustes no tratamento dado aos textos, de maneira a incrementar o desempenho do modelo.

Após obtida acurácia satisfatória e não ser possível maiores melhorias, é consolidada a última etapa do fluxo da figura 6. Nessa é realizada a persistência do melhor modelo alcançado, de maneira a serem geradas 3 estruturas que compõem o modelo:

- Estrutura representando a **frequência** de ocorrência das palavras nas mensagens (*bag of words*);

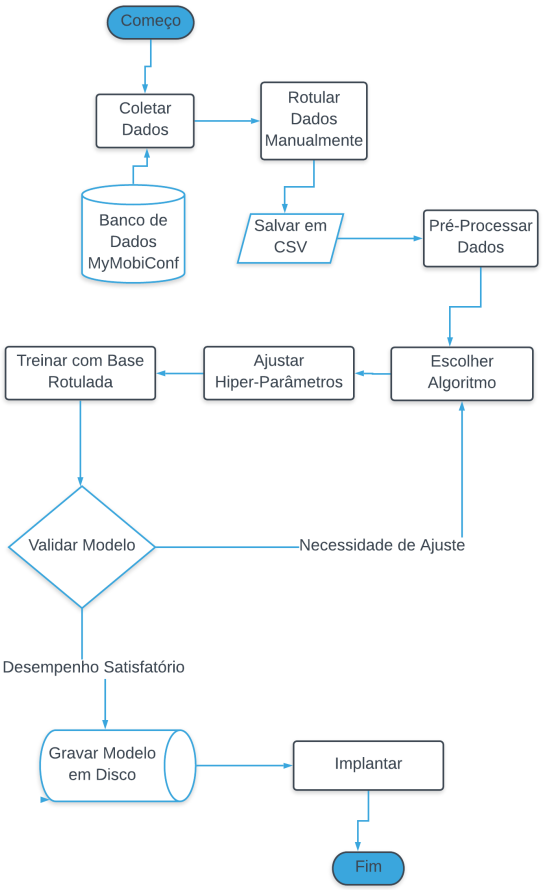

Figura 7. Fluxograma para Desenvolvimento do Modelo Apresentado

- Estrutura representando a **tabela de probabilidade**, construída pelo Teorema de Bayes, a partir da estrutura anterior;
- Estrutura representando o **modelo** em si, construído a partir das duas estruturas anteriores.

Essas são gravadas como arquivos binários em disco, a fim de que o mesmo, já construído e treinado, fique armazenado permanentemente, de forma que nas próximas execuções do programa para a análise de sentimento, o modelo seja carregado diretamente de onde se encontra, sem necessidade de se repetir todas as fases da sua criação.

A figura 7 ilustra o passo-a-passo descrito para criação do modelo apresentado nesse trabalho. Como visto, após a persistência, é feita a implantação do modelo, o integrando à versão *web* do MyMobiConf, como um módulo de software a ser executado. Desse modo, será responsável por efetuar a análise das opiniões de eventos cadastrados, à medida que novas opiniões e seus respectivos eventos passem a fazer parte da base de dados. O resultado da classificação será exibido apenas para os usuários cadastrados como organizadores do evento, os quais possuem acesso às suas estatísticas.

5. Discussão de Resultados

Nesta Seção, são apresentados os resultados após efetuada a construção do modelo de classificação, com base na validação efetuada e métricas consideradas, a fim de se verificar

seu desempenho, mostrando também seu uso integrado à versão web do MyMobiConf para classificação de opiniões de determinados eventos.

É de suma importância avaliar a qualidade de modelos de classificação, sendo que existem diversas métricas com propósitos e características diferentes, com o intuito de validar se o modelo criado com base nos dados disponíveis é capaz de realizar boas previsões para dados desconhecidos. Uma das formas mais confiáveis para essa validação - e que é empregada nesse trabalho - é a validação cruzada.

A técnica consiste em dividir o conjunto de dados em três partes:

- **Treinamento**: Usada para treinar o modelo;
- **Validação**: Usada para ajustar os parâmetros;
- **Teste**: Usada ao final, como teste em dados desconhecidos.

A figura 8 ilustra essa divisão. Como visto, o conjunto é inicialmente dividido em treino e teste, de maneira que os dados de teste serão desconhecidos para o classificador. Então, com os dados de treino, é feita a validação do modelo para diferentes sub-conjuntos desses dados, de maneira que são executadas n rodadas, sendo, em cada uma, feita a sub-divisão do conjunto em treino e validação, com o objetivo de, em cada uma dessas execuções, treinar o modelo e validá-lo para diferentes combinações dos dados. Ao fim, calcula-se a média de desempenho das várias rodadas.

Esse mecanismo evita fixar conjuntos para treino e teste, a fim de se aproveitar ao máximo os dados para refinar o modelo. Após essa validação e obtenção do modelo de melhor desempenho, é feita a aplicação do mesmo sobre o conjunto de teste, para que seja comprovada sua eficácia, e consequente capacidade de generalização, funcionando bem também em dados novos.

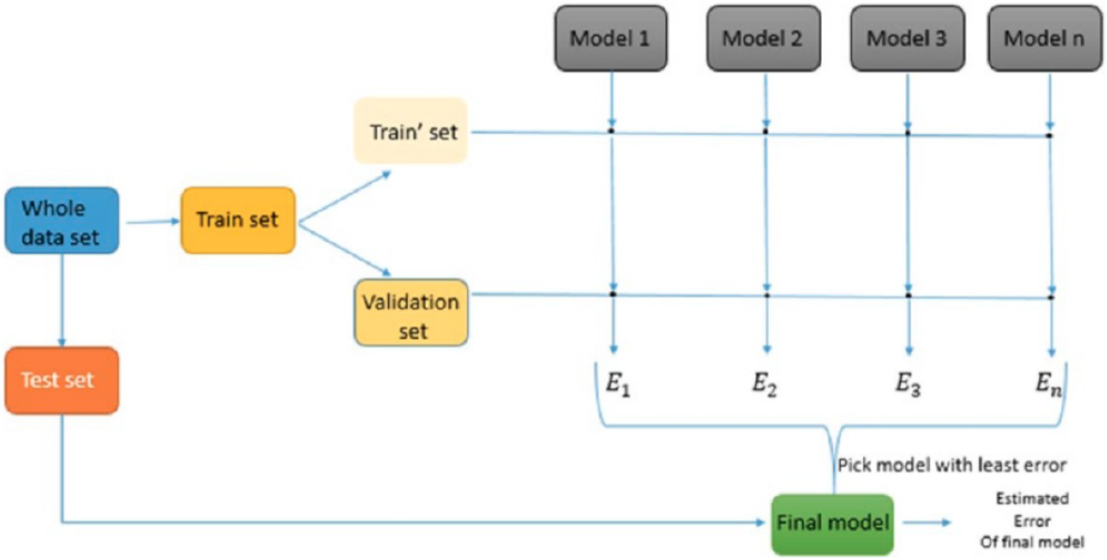

Figura 8. Esquema Representativo para a Validação Cruzada - Fonte: Practical Machine Learning with Python (Dipanjan Sarkar, Raghav Bali e Tushar Sharma, 2018)

Para o modelo de classificação desenvolvido, foi utilizada a validação cruzada

da própria *scikit-learn*, a qual realiza automaticamente o procedimento descrito para um número definido de execuções, além de já calcular as principais métricas de desempenho para o conjunto de execuções. Foi definida a validação para dez sub-conjuntos, considerando as métricas descritas a seguir:

- **Acurácia**: Proporção de classificações corretas do modelo, sendo muito empregada quando as classes envolvidas são balanceadas e possuem a mesma relevância;
- **Precisão**: Determina, para as previsões feitas para uma classe, a proporção das corretas, tendo foco nessa classe;
- **Revocação**: Determina, para determinada classe presente no conjunto de dados, a proporção daquelas previstas corretamente. Normalmente seu aumento pode levar a uma redução da precisão e vice-versa;
- **F1 Score**: Média harmônica da precisão e revocação, sendo importante para o balanceamento entre ambas.

Enquanto a acurácia mede o desempenho geral do modelo, as demais métricas trazem o desempenho específico para as classes envolvidas. De modo geral, tratam de medir a taxa de acertos e erros para cada classe considerada, avaliando a capacidade do algoritmo de prever corretamente uma classe ou rotulá-las erroneamente. Por sua vez, a *Macro-Average* é uma medida adicional, que computa a métrica independente para cada classe e calcula a média (consequentemente todas as classes são igualmente tratadas), sendo preferível para tratar poucas classes.

A tabela 1 exibe os melhores resultados após a validação do modelo de acordo com o procedimento descrito anteriormente, tendo a configuração de hiper-parâmetros, além das modificações no tratamento e representação das mensagens do conjunto de treino que proporcionaram melhor desempenho. Já a figura 9, mostra a Matriz de Confusão, um formato popular para avaliação das classificações feitas por um modelo, além de ser uma excelente forma de visualizar as proporções de erros e acertos das previsões. Basicamente, compara os rótulos reais com os obtidos pelo algoritmo para cada classe. A mesma foi construída em forma de *Heatmap*, a fim de se ter melhor visualização (cores mais fortes indicam maiores ocorrências), de modo que os valores maiores encontram-se na forma de potência. A diagonal principal indica a quantidade de acertos de cada classe, já as demais células mostram o número de erros de previsão de uma classe, rotulada equivocadamente como outra.

Tabela 1. Métricas de Desempenho da Validação Cruzada para o Classificador

	Precisão	**Revocação**	**F1 Score**
Positivo	67%	49%	56 %
Negativo	50%	4%	8%
Neutro	64%	87%	74%
Macro-Average	60%	47%	46%

A acurácia relacionada às métricas da tabela foi de **64.58%** para a validação. Ademais, foi feito o teste do modelo para o conjunto de dados desconhecido por esse, separado da validação e treino, obtendo-se uma acurácia consideravelmente próxima de **62%**.

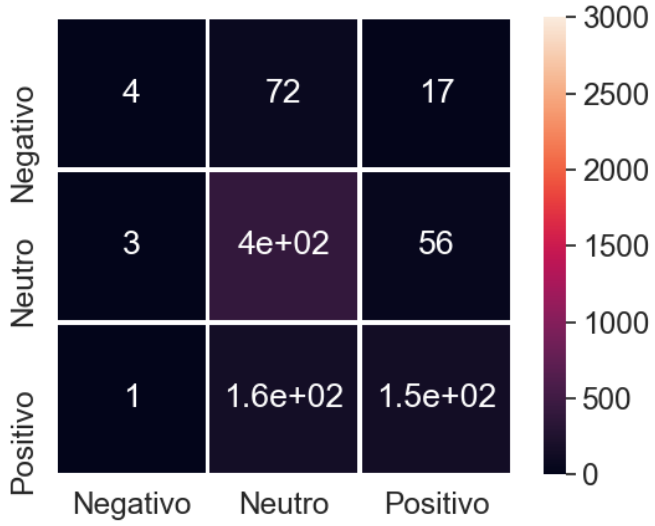

Figura 9. Matriz de Confusão para o Classificador

Estes resultados são satisfatórios, levando-se em conta o contexto e a limitação da base de dados utilizada para o processo de treinamento do modelo, assim como a maior dificuldade acrescentada ao se considerar análise de textos em português. Além disso, a habilidade do ser humano de avaliar corretamente emoções presentes em formas textuais situa-se na faixa de 72% a 85% [16]. Dessa forma, dada a dificuldade de se tratar a subjetividade através da análise de sentimento feita desempenhada por um algoritmo, juntamente das limitações de recursos para aplicação da mesma no cenário considerado, foi obtido desempenho suficientemente bom pelo método adotado para classificação.

Pode-se notar, pela matriz de confusão, que há um desbalanceamento entre as taxas de acerto para as classes, sendo que, para mensagens neutras, os acertos foram elevados, enquanto que para as positivas houve um equilíbrio entre erros e acertos, já para as negativas houve uma taxa de erros elevada. Tal resultado se deve ao fato de que, para o treino do classificador, houve uma grande variação do número de mensagens entre classes, logo o mesmo teve uma forte base para classes neutras e poucos exemplos para reconhecimento de sentenças negativas. Dessa forma, tende a classificar equivocadamente frases negativas como neutras (célula de valor 72 da Matriz).

Analisando as métricas obtidas para a polaridade negativa, comprova-se o desempenho inferior das predições para a mesma, devido ao F1-Score muito baixo, além de que a revocação ainda menor indica que as predições feitas para essa classe, em sua maioria, foram errôneas. Porém, como a maioria dessas previsões incorretas foram considerando mensagens negativas como neutras, tem-se um cenário relativamente melhor se o equívoco fosse com positivas, o que é reforçado pelo fato de que é difícil se atingir um consenso na literatura para definição de características de neutralidade em uma mensagem [14].

O desequilíbrio entre as classes também afetou a polaridade positiva, visto que houve grande classificação das mesmas como neutras, porém também há muitos acertos, já que há um número razoável de mensagens positivas presentes na base de dados, de

tal maneira que as métricas demonstram um desempenho razoável, sendo a maioria das classificações feitas corretamente para essa classe. Já em relação à polaridade neutra, quase todas as predições para a mesma foram corretas, sendo muito bem distinguidas das demais classes, como visto na matriz de confusão. Além disso, as métricas demonstram principalmente que, das mensagens neutras presentes no conjunto de dados, a maioria foi prevista corretamente (alta revocação).

A partir desse cenário, confirma-se como a qualidade do conjunto de dados de treino afeta o desempenho do classificador, sendo o principal empecilho para sua melhoria no presente trabalho.

Ainda assim, considerando que foi obtida uma taxa de acertos suficientemente boa, dadas as características do problema, pode-se deduzir que, entre os participantes dos diversos eventos registrados no MyMobiConf, há uma maioria que não se mostra muito engajada nos mesmos ou interessadas a ponto de sentirem impactos beneficentes advindos dessas atividades, devido à maior neutralidade. Entretanto, vale ressaltar que muitas opiniões dessa polaridade podem designar críticas construtivas e apoio aos eventos e suas atividades, sem necessariamente expressar algo claramente positivo, que seria assim classificado pelo algoritmo. Unindo-se isto ao número considerável de mensagens positivas, pode-se dizer que há boa parcela dos participantes que demonstra interesse nos eventos realizados e se sentem gratificados pelos mesmos. Há também uma quantidade reduzida de críticas negativas, o que mostra que, na visão de uma maioria, os eventos possuem poucos pontos que possam gerar desconforto e insatisfação.

Por fim, foi feita o cálculo da curva ROC (*Receiver Operating Characteristic*, ou *Características Operacionais do Receptor*) e da AUC (*Area under the Curve*, ou *Área sob a Curva*) correspondente para cada classe, levando-se em conta as predições do modelo, como mostrado nas figuras 10, 11 e 12. Essa mostra a capacidade de o modelo distinguir entre as classes, de acordo com a probabilidade usada para definir a qual classe uma instância pertence (Probabilidade depende de cada algoritmo). A curva é definida separadamente para cada classe, visto que deve-se considerar instâncias classificadas corretamente como sendo ou não da classe (Verdadeiro Positivo, Falso Positivo). A diagonal representa um classificador aleatório, logo, quanto mais a curva do classificador estiver acima dessa, melhor o mesmo será. Além disso, a Área sob a Curva (AUC) é um valor numérico que representa o desempenho do classificador (quanto maior, melhor, sendo 1 o melhor caso). Por assim ser, é uma maneira útil e visualmente compreensível para verificar o desempenho de um modelo.

Vale ressaltar que o resultado do cálculo das curvas demonstra uma alta precisão para todas as classes, muito superior às métricas da Tabela 1, contudo, é considerada uma binarização das 3 classes, levando-se em conta a probabilidade de uma instância pertencer ou não a uma classe, como se houvesse apenas duas classes, o que justifica o incremento da precisão. Tal fato acarreta que, se fossem utilizadas apenas as polaridades Negativo e Positivo, o desempenho do modelo seria muito superior. Dado isto, a área tão próxima entre as classes se justifica pelo fato de que, para cálculo dessa, não há o peso do desbalanceamento provocado pela neutralidade, que foi o maior fator responsável pelo declínio das métricas para as demais.

Como etapa final do trabalho que desenvolvemos, após feitas as análises de desem-

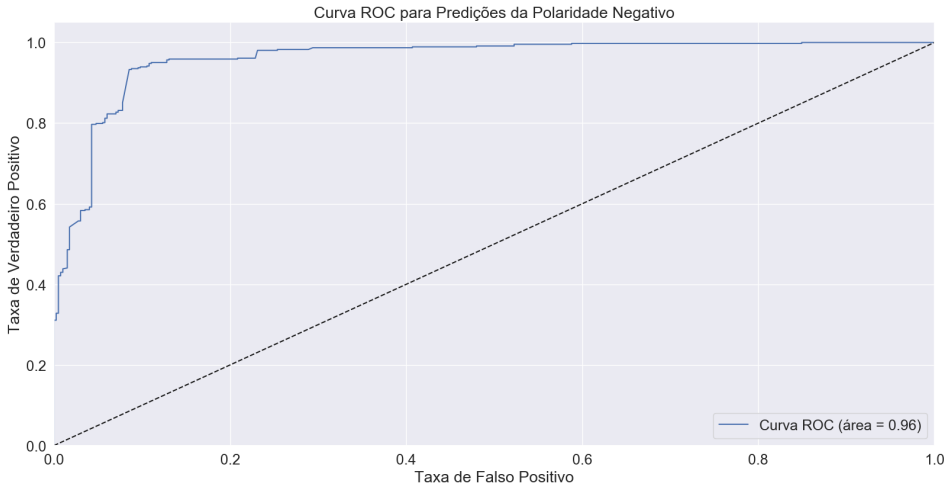

Figura 10. Curva ROC e AUC calculados para a Polaridade Negativo

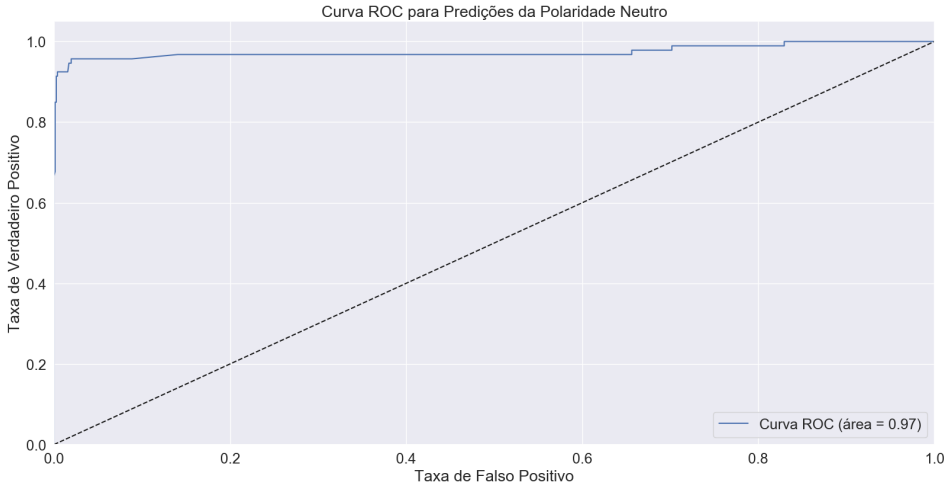

Figura 11. Curva ROC e AUC calculados para a Polaridade Neutro

penho e obtidos resultados satisfatórios, o modelo foi integrado à versão *web* do MyMobi-Conf, como um módulo adicional executado separadamente. Dessa forma, quando algum organizador de um evento cadastrado acessar a página desse e direcionar-se até sua seção de opiniões, na qual são exibidas algumas estatísticas acerca das mesmas, será também exibida uma tela como das figuras 13 e 14, que irá exibir a quantidade de opiniões, para aquele evento, pertencente a cada polaridade, bem como a polaridade predominante para todo o conjunto de mensagens. Ademais, são mostrados ícones representativos para cada polaridade, sendo destacado aquele da predominante. As telas das figuras 13 e 14 correspondem, respectivamente, aos eventos VII SECOM (Semana da Computação) e VIII SECOM (Semana da Computação), ambos realizados no campus Florestal da UFV nos anos de 2018 e 2019. O resultado confirma as discussões feitas anteriormente.

Essa visualização está disponível apenas para organizadores e vale para qualquer

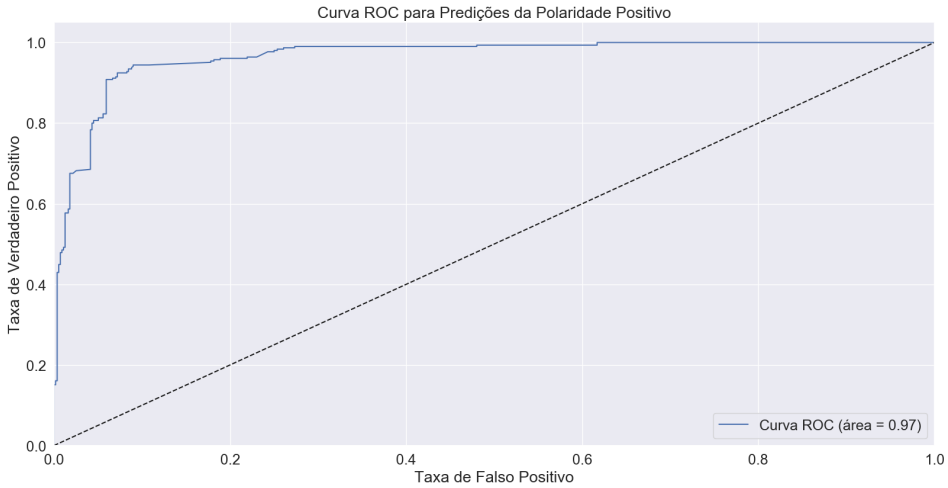

Figura 12. Curva ROC e AUC calculados para a Polaridade Positivo

que seja o evento registrado, que possua qualquer número de opiniões. Ademais, é importante ressaltar que, à medida que novas opiniões são adicionadas na seção de um evento, o modelo as classifica e recalcula a polaridade predominante. Para tanto, não é necessário reavaliar mensagens já classificadas, tão pouco reconstruir o modelo, esse é apenas carregado do disco e executado. Adicionalmente, foi elaborado um *script* para verificar a presença de novas mensagens em certo período de tempo, e então executar o modelo quando necessário.

Figura 13. Polaridade das Opiniões da VII SECOM - Modelo Integrado ao MyMobiConf Web

Figura 14. Polaridade das Opiniões da VIII SECOM - Modelo Integrado ao MyMobiConf Web

6. Considerações Finais

No presente trabalho, foi desenvolvido um método para a análise de sentimento, a fim de se extrair a polaridade expressa por mensagens do idioma português, tendo como foco as opiniões de usuários do aplicativo MyMobiConf, em relação aos eventos de que participaram. Para tanto, foi construído um modelo de classificação, por meio de aprendizado supervisionado, baseado no Teorema de Bayes. Para treinamento do classificador, utilizou-se a base de dados do MyMobiConf referente às mensagens dos usuários, para o período de 2016 a 2019, as quais foram rotuladas manualmente como positivas, negativas ou neutras. Esses dados rotulados foram pré-processados, segundo procedimento apresentado, com a finalidade de torná-los o mais adequados possíveis para serem utilizados pelo algoritmo de classificação. Posteriormente, tais dados foram divididos em conjuntos de treino, validação e teste, para que fosse aplicada a técnica de validação cruzada, segundo as principais métricas de desempenho, para avaliação do desempenho do modelo e realização dos ajustes necessários.

Resultados mostram-se consideravelmente bons, dada a dificuldade envolvida na tarefa de análise da subjetividade, mesmo por pessoas, bem como levando-se em conta o contexto em questão e suas limitações. Desse modo, foi identificado que os maiores empecilhos se referem à quantidade relativamente menor de mensagens para treinamento, junto do desbalanceamento do número de mensagens por polaridade. Tal fato impactou na grande diferença entre as métricas de desempenho para as predições do modelo para cada classe, de maneira que a classe Neutro apresentou resultados muito melhores, ficando a classe Positivo com resultados intermediários e a classe Negativo desfavorecida, além de haver o maior potencial do classificador no caso de se envolver apenas duas polaridades. Por fim, observou-se a maior tendência de neutralidade dos usuários para com os eventos e um número reduzido de críticas negativas, o que junto ao fato de se ter um número considerável de opiniões positivas, demonstra que há uma parcela relativamente alta dos participantes que se mostra muito interessada nas atividades realizadas, havendo potencial para engajar um maior número de indivíduos.

Pode-se dizer que esse trabalho se destaca por ser uma das poucas aplicações da análise de sentimento envolvida no contexto de eventos que também englobam o ambiente acadêmico, considerando-se a Língua Portuguesa, além de trazer, através de um sistema

web, os resultados de seu emprego, sendo esses visualmente chamativos aos organizadores dos eventos. Para trabalhos futuros, objetiva-se elevar o desempenho do classificador para o mesmo contexto, o que pode ser alcançado através da ampliação da base de dados, rotulada, usada para treinamento, bem como do balanceamento da quantidade de mensagens entre as polaridades tratadas, juntamente do uso de textos com conteúdo diversificado, vocabulário variado e com ruídos reduzidos, o que possibilita que o modelo conheça um maior número de situações para cada polaridade e se torne mais apto a distinguir os sentimentos. Ademais, é pensado no uso de técnicas de aprendizado de máquina mais eficazes, como *deep learning*.

Referências

[1] L. K. da Silva, M. L. K. Barbosa, R. Pandolfi, and S. C. Cazella, "Análise de sentimento pela ótica da abordagem multimodal," *RENOTE*, vol. 15, no. 1.

[2] B. Liu, "Sentiment analysis and opinion mining," *Synthesis lectures on human language technologies*, vol. 5, no. 1, pp. 1–167, 2012.

[3] F. Benevenuto, F. Ribeiro, and M. Araújo, "Métodos para análise de sentimentos em mídias sociais," in *Brazilian Symposium on Multimedia and the Web (Webmedia), Manaus, Brasil*, 2015.

[4] N. F. F. d. Silva, *Análise de sentimentos em textos curtos provenientes de redes sociais*. PhD thesis, Universidade de São Paulo, 2016.

[5] J. A. CARVALHO FILHO, "Mineração de textos: Análise de sentimento utilizando tweets referentes à copa do mundo 2014," 2014.

[6] R. L. Rosa, *Análise de sentimentos e afetividade de textos extraídos das redes sociais.* PhD thesis, Universidade de São Paulo, 2015.

[7] E. J. de Aguiar, B. S. Faiçal, J. Ueyama, G. C. Silva, and A. Menolli, "Análise de sentimento em redes sociais para a língua portuguesa utilizando algoritmos de classificação," in *Anais do XXXVI Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos*, SBC, 2018.

[8] V. L. S. de Lima, M. d. G. V. Nunes, and R. Vieira, "Desafios do processamento de línguas naturais," *SEMISH-Seminário Integrado de Software e Hardware*, vol. 34, p. 1, 2007.

[9] P. F. Matos12, "Metodologia de pré-processamento textual para extração de informação em artigos científicos do domínio biomédico," in *VIII Workshop de Teses e Dissertações em Banco de Dados*, 2009.

[10] C. J. Hutto and E. Gilbert, "Vader: A parsimonious rule-based model for sentiment analysis of social media text," in *Eighth international AAAI conference on weblogs and social media*, 2014.

[11] J. Reis, P. Gonçalves, M. Araújo, A. C. Pereira, and F. Benevenuto, "Uma abordagem multilíngue para análise de sentimentos," in *Anais do IV Brazilian Workshop on Social Network Analysis and Mining*, SBC, 2015.

[12] G. Lucca, I. A. Pereira, A. Prisco, and E. N. Borges, "Uma implementaçao do algoritmo naıve bayes para classificaçao de texto," *Centro de Ciências Computacionais-Universidade Federal do Rio Grande (FURG)*, 2013.

[13] P. Nascimento, R. Aguas, D. De Lima, X. Kong, B. Osiek, G. Xexéo, and J. De Souza, "Análise de sentimento de tweets com foco em notícias," in *Brazilian Workshop on Social Network Analysis and Mining*, 2012.

[14] T. C. de França and J. Oliveira, "Análise de sentimento de tweets relacionados aos protestos que ocorreram no brasil entre junho e agosto de 2013," in *Proceedings of the III Brazilian Workshop on Social Network Analysis and Mining (BRASNAN)*, pp. 128–139, 2014.

[15] J. H. Martin and D. Jurafsky, *Speech and language processing: An introduction to natural language processing, computational linguistics, and speech recognition*. Pearson/Prentice Hall Upper Saddle River, 2009.

[16] J. Wiebe, T. Wilson, and C. Cardie, "Annotating expressions of opinions and emotions in language," *Language resources and evaluation*, vol. 39, no. 2-3, pp. 165–210, 2005.