UM SOFTWARE PARA A CLASSIFICAÇÃO DAS AÇÕES GERENCIAIS DOS EMPREGADOS DA EMBRAPA UTILIZANDO TÉCNICAS DE MINERAÇÃO DE TEXTO

Ana Cláudia Alves Mendes Araújo e Cristiano Francis Matos de Macedo

Abstract— **Este trabalho apresenta um estudo de caso de mineração de texto para classificar automaticamente as atividades desempenhadas pelos empregados da Embrapa. Essas são descritas textualmente e são de preenchimento livre. Elas são cadastradas pelo superior imediato do empregado que muitas vezes comete o erro de cadastrá-las com o tipo errado. Este trabalho irá aplicar técnicas de mineração de texto para auxiliar na classificação dessas atividades. Compararemos quatro algoritmos de modelagem; Naives Bayes, Árvore de Decisão, Floresta Aleatória e SVM; de modo a escolher o que apresentar o melhor resultado. Ao final apresentaremos o software desenvolvido que realiza a classificação.**

I. INTRODUÇÃO

A Embrapa (Empresa Brasileira de Pesquisa Agropecuária) [1] é uma empresa pública do governo federal que tem como objetivo viabilizar soluções de pesquisa, desenvolvimento e inovação para a sustentabilidade da agricultura, em benefício da sociedade brasileira[1].

Atualmente a Embrapa conta com cerca de 10 mil empregados. Anualmente é feito o planejamento das atividades que serão realizadas por eles. As atividades podem ser classificadas em dois tipos: atividades de projetos ou ações gerenciais (atividades gerenciais). A descrição das atividades é um campo do tipo texto não estruturado.

As atividades de projeto são atividades vinculadas ao planejamento, execução e controle dos projetos realizados na Embrapa. Geralmente estão ligadas às atividades da área fim da Empresa, a pesquisa. Já as ações gerenciais são atividades realizadas como suporte para as demais áreas e processos da empresa.

O objetivo deste trabalho é utilizar a mineração de texto para criar um aplicativo capaz de classificar as atividades dos empregados, este aplicativo poderá ser utilizado em dois momentos, antes do cadastro da atividade, como uma forma de validação prévia da classificação da atividade, ou a posteriori no processo de revisão como uma forma de encontrar possíveis erros de classificação nas atividades já cadastradas. Esse último é um processo manual sujeito a erro e com um consumo muito grande de tempo do revisor.

Para alcançar este objetivo propomos a utilização das técnicas de mineração de texto e aprendizagem de máquina para extração de padrões das bases de dados de ações gerenciais e de atividades de projetos [2] a fim de criar um modelo capaz de classificar automaticamente as atividades.

[1]www.embrapa.br

A seção II apresenta os trabalhos correlatos que utilizamos como base para o uso da mineração de texto. A seção III apresenta a metodologia utilizada na mineração dos dados. As seções de IV à VIII apresentam as fases da metodologia adotada. A seção IX traz o aplicativo desenvolvido. e por fim a seção X apresenta as conclusões e sugestões de trabalhos futuros.

II. TRABALHOS CORRELATOS

Apresentamos nessa seção dois exemplos de uso da mineração de textos para classificação de documentos que se assemelham com a proposta desse artigo.

No primeiro caso, [3], foi feita a classificação de queixas cadastradas pelos cidadãos com relação às denúncias de irregularidades no uso dos recursos públicos. O modelo proposto foi construído utilizando a mineração de texto. Foram aplicados os seguintes algoritmos de aprendizagem de máquina: Máquina de Vetores de Suporte - SVM, Naive Bayes, Floresta Aleatória e Árvore de Decisão. O algoritmo que obteve o melhor resultado foi o Floresta Aleatória. Os resultados obtidos mostraram que é possível implementar um classificador automático usando mineração de texto para triagem das reclamações.

No segundo caso, [4], foi feita a identificação de relações de formulários de registro numa base de dados, começando com a mãe, porque esse papel é a base relacional de vários outros. As informações de relações entre pessoas é muito importante para diversas áreas. No cenário de investigação criminal que desempenha um papel fundamental na compreensão de como pessoas podem influenciar outras e ser influenciadas. Resumindo os dados de uma proposta na qual a informação é recolhida de duas pessoas e processados com o objetivo de atribuir uma pontuação de similaridade de nome e endereço. Em seguida, aplica-se a aprendizagem de máquina para classificar a relação dessas pessoas utilizando dois algoritmos: Máquina de Vetores de Suporte - SVM, com base em hiperplanos e Naive Bayes, um modelo que usa probabilidade condicional para classificar a entrada. Inicialmente, fizeram uso de vários atributos. Posteriormente, verificaram que o modelo que apresentava o melhor desempenho era o SVM.

Além dos trabalhos citados, Floriano e Kaestner [5] apresentam um trabalho sobre a utilização do R para realização de mineração de dados, neste artigo realizamos a modelagem dos dados utilizando essa ferramenta estatística.

III. Metodologia

O Processo Padrão Inter-Indústrias para Mineração de Dados, também conhecido pela sigla CRISP-DM (*Cross Industry Standard Process for Data Mining*) [6], é um modelo de processo de mineração de dados que descreve as abordagens mais usadas por especialistas para mineração de dados.

Esse processo possui em seu ciclo de desenvolvimento as seguintes fases: entendimento do negócio, entendimento dos dados, preparação dos dados, modelagem, avaliação dos dados e desenvolvimento.

Cada fase tem suas atribuições como descrito abaixo:

- Entendimento do Negócio: essa fase foca em entender o objetivo do projeto a partir de uma perspectiva de negócio, definindo um plano preliminar para atingir o objetivos.
- Entendimento dos Dados: nessa fase é feito o recolhimento dos dados e o início das atividades para familiarização com os dados, identificando problemas ou conjuntos interessantes.
- Preparação dos Dados: fase responsável pela construção dos dados finais a partir dos dados iniciais. Geralmente ocorre várias vezes no processo.
- Modelagem: nessa fase são aplicadas várias técnicas de modelagem, e seus parâmetros são calibrados para otimização. Assim, é comum retornar à fase de Preparação dos Dados durante essa fase.
- Avaliação dos Dados: fase onde é construído um modelo que parece ter grande qualidade de uma perspectiva de análise de dados. No entanto, é necessário verificar se o modelo atinge os objetivos do negócio.
- Desenvolvimento: nessa fase o conhecimento adquirido pelo modelo é organizado e apresentado de maneira que o cliente possa utilizar.

A fase de entendimento do Negócio foi descrita na seção de introdução e segue uma complementação na seção seguinte. As demais fases de Entendimento dos Dados, Preparação dos Dados, Modelagem, Avaliação dos Dados e Desenvolvimento estão detalhadas nas próximas seções.

IV. Entendimento do Negócio

Complementando o que já foi dito na introdução, para um melhor esclarecimento do contexto do problema a ser resolvido, apresentamos na Tabela I exemplos de texto descritivos das atividades de projeto e das ações gerencias. Podemos perceber que nas atividades gerenciais o termo Gestão aparece em três exemplos, já o termo elaboração aparece duas vezes nas atividades de projeto e nenhuma vez nas ações gerenciais. Obviamente, este é um exemplo resumido, o que se espera é que os algoritmos de aprendizagem de máquina consigam classificar corretamente os conjuntos de dados apresentados neste trabalho.

V. Entendimento dos Dados

Os dados utilizados neste trabalho foram extraídos diretamente da base de dados do sistema de gestão estratégica

TABLE I
EXEMPLO DE ATIVIDADES

Atividades de Projeto
Workshop final do projeto
Análise de dados e teste de hipótese
Construção de uma homepage
Elaboração de relatórios técnicos do projeto
Elaboração de relatório
Ações gerencias
Gestão e Controle
Coordenação do CLPI
Gestão do Campo Experimental
Gestão do desempenho
Coordenação das ações de coleta de lixos da Unidade

e do sistema de gerenciamento de projetos da Embrapa. O primeiro sistema é responsável pelo cadastramento das ações gerenciais dos empregados, o segundo é responsável pelas atividades de projeto atribuídas a cada empregado. Os dois sistemas são interligados.

Para este trabalho foi selecionado apenas o campo de dados com a descrição das atividades.

Ainda como atividade realizada para entendimento dos dados, foram feitas diversas análises exploratórias dos dados coletados, uma das informações que utilizamos para entender melhor as características específicas de cada tipo de atividade, foi o levantamento da frequência dos termos.

A Figura 1 traz os 10 termos com maior frequência para o tipo atividade de projeto, já a Figura 2 traz os 10 termos com maior frequência para o tipo de atividade ação gerencial. Podemos observar uma diferença entre os termos que mais aparecem.

Fig. 1. Termos mais frequentes das atividades de projeto.

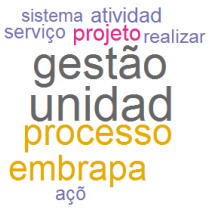

Fig. 2. Termos mais frequentes das ações gerenciais.

VI. Preparação dos Dados

O primeira passo para a preparação dos dados foi a exportação dos dados das bases de dados transacionais para

o formato de arquivo CSV [2], como utilizamos somente um dos campos que descrevem a atividades, o campo descrição, geramos artificialmente um outro campo contendo o tipo da atividade, se essa era uma atividade de projeto ou ação gerencial. Este campo gerado será utilizado na modelagem para que o algoritmo possa extrair os padrões. Os dados foram exportados utilizando o aplicativo Squirell [3].

Optamos por separar em dois arquivos, um para as ações gerencias e o outro para as atividades de projeto. Não realizamos nenhuma filtragem dos dados durante a preparação dos dados. Por questões de sigilo reduzimos o número de amostras, sendo 4000 amostras de cada tipo de atividade. Tivemos problemas com o uso dos dados exportados devido a codificação do campo texto, de forma que padronizamos que todos os dados exportados seriam convertidos para padrão de codificação de caractere UTF-8

VII. Modelagem

Nesta etapa utilizamos o RStudio [4] para realizar a modelagem. Como primeiro passo dividimos o conjunto de dados em três bases de dados. A primeira contendo 60% dos dados será utilizada para o treinamento dos modelos, a segunda contendo 20% dos dados será utilizada para validar e comparar os modelos, e a última contendo 20% dos dados será utilizada para testar se o modelo escolhido pode ser generalizado pelos casos não vistos. Nesta divisão utilizamos a função *sample* do R para garantir a aleatoriedade dos dados, bem como a função *set.seed* para depois podermos reproduzir os testes.

Após a divisão das amostras, utilizamos a biblioteca *tm* do R para a extração dos atributos do texto e para o treinamento dos modelos utilizamos as bibliotecas rparty(árvore de decião), randomForest (floresta aleatória) e a biblioteca 'e1071' (naive Bayes e SVM).

Durante o processo de extração dos atributos do texto aplicamos as seguintes transformações: transformando todo o texto para minúsculo, removemos os números e pontuação, removemos as *stopwords* da língua portuguesa e reduzimos os termos às suas raízes (*stemming*) de modo a termos o radical de cada termo extraído do texto.

Utilizamos a função *DocumentTermMatrix* para construir a matriz de termos a partir do texto, e aplicamos a funcão *removeSparseTerms* para remover as palavras com pouca ocorrência. Antes da remoção, tínhamos 6523 termos, após a remoção, o número de termos foi reduzido para 119. Depois desta etapa, passamos para a etapa de modelagem propriamente dita em que foram aplicados os algoritmos de treinamento propostos neste trabalho. Após a extração dos modelos, avaliamos a performance dos modelos criados extraindo a matriz de confusão e junto com outras informações estatísticas extraídas com a biblioteca *caret* do R.

[2]Comma-separated values
[3]http://squirrel-sql.sourceforge.net/
[3]https://pt.wikipedia.org/wiki/UTF-8
[4]https://www.rstudio.com/

O script utilizado para treinamento está disponibilizado no github [5]

VIII. Avaliação dos dados

Nesta fase, iremos identificar o melhor modelo para o problema proposto neste artigo. Para isso analisaremos as informações geradas na matriz de confusão e nas informações estatísticas extraídas desta.

As tabelas II,III, IV e V trazem a matriz de confusão para cada algoritmo, nestas é possível observar o comportamento dos algoritmos em relação ao número de falso positivos (FP), falso negativos (FN), verdadeiros positivos (TP) e verdadeiros negativos T(N) [2].

TABLE II
MATRIZ DE CONFUSÃO PARA ÁRVORE DE DECISÃO

ÁRVORE DE DECISÃO	AÇÃO GERENCIAL	PROJETO
AÇÃO GERENCIAL	506	294
PROJETO	67	733

TABLE III
MATRIZ DE CONFUSÃO PARA NAIVE BAYES

NAIVE BAYES	AÇÃO GERENCIAL	PROJETO
AÇÃO GERENCIAL	535	265
PROJETO	113	687

TABLE IV
MATRIZ DE CONFUSÃO PARA FLORESTA ALEATÓRIA

ÁRVORE ALEATÓRIA	AÇÃO GERENCIAL	PROJETO
AÇÃO GERENCIAL	631	169
PROJETO	76	724

TABLE V
MATRIZ DE CONFUSÃO PARA O SVM

SVM	AÇÃO GERENCIAL	PROJETO
AÇÃO GERENCIAL	632	168
PROJETO	84	716

Apartir da tabela de decisão importantes informações podem ser extraídas, neste trabalho utilizaremos as seguintes informações para podermos escolher o melhor algoritmo: Acurácia, Sensibilidade (*Sensitivity*) e Especifidade (*Specificity*).

A Acurácia é descrita na equacão 1 e denota a proporção total de classificações corretas. Especificidade é a taxa de verdadeiros negativos é descrita na equação 2. Sensibilidade mede a taxa de verdadeiros positivos e é descrita na equação 3.

$$Acuracia = \frac{Tp + Tn}{Tp + Fn + Fp + Tn} \quad (1)$$

[5]https://github.com/cmacedo80/mestradocfmmdm

$$Especificidade = \frac{Tp}{Tp + Fn} \quad (2)$$

$$Sensibilidade = \frac{Tn}{Tn + Fp} \quad (3)$$

A Tabela VI mostra os resultados estatísticos de cada algoritmo utilizado. De acordo com os resultados obtidos o algoritmo que apresentou o melhor resultado foi o de floresta aleatória, seguido do algoritmo SVM. Os dois apresentaram resultados muito semelhantes.

TABLE VI
ACURÁCIA, SENSIBILIDADE E ESPECIFICIDADE PARA CADA ALGORITMO

Algoritmo	Acurácia	Sensibilidade	Especificidade
Árvore de decisão	0.7744	0.8831	0.7137
Naive Bayes	0.7638	0.8256	0.7216
Floresta Aleatória	0.8469	0.8925	0.8108
SVM	0.8425	0.8827	0.8100

Com o intuito de validar esse resultado. Avaliamos o melhor modelo com a base de teste, os resultados podem ser vistos na tabela VII. Como os resultados foram semelhantes ao obtido com a base de teste, podemos inferir que o modelo pode ser generalizado para dados nunca vistos.

TABLE VII
ACURÁCIA, SENSIBILIDADE E ESPECIFICIDADE PARA PARA A BASE DE TESTE

Algoritmo	Acurácia	Sensibilidade	Especificidade
Floresta Aleatória	0.8225	0.8644	0.7892

IX. Aplicativo Desenvolvido

Para o desenvolvimento do aplicativo, tivemos que considerar os seguintes requisitos: Primeiro, o software deveria comunicar-se com o R a fim de realizar a predição; Segundo, o software deveria ser desenvolvido em uma linguagem multiplataforma de modo que pudesse ser executado em vários sistemas operacionais. Diante desses requisitos, foi escolhida a linguagem de desenvolvimento Java [6], utilizando a biblioteca de RServe [7] para realizar a comunicação com o R.

A Figura 3 mostra a tela do software desenvolvido. O funcionamento do software é simples. O usuário digita no campo o texto a ser classificado e pressiona o botão Processar, assim que o botão é pressionado o software realiza uma chamada para o R que executa o script de predição de texto e retorna para o software a classificação encontrada para o texto. Importante salientar que, o software permite realizar a classificação de várias atividades, basta informar uma atividade por linha.

[6]http://www.java.com
[7]https://rforge.net/Rserve/

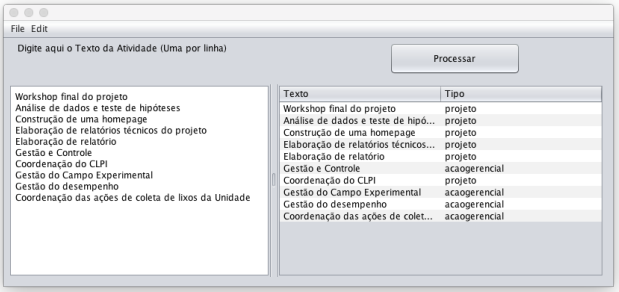

Fig. 3. Tela do software desenvolvido.

Todo o código fonte do software está disponibilizado publicamente em um repositório público do github[8], além do código fonte estão disponibilizados os scripts em R. A base de dados por questão de sigilo não foi disponibilizada.

X. Conclusão e Trabalhos Futuros

Este trabalho foi uma prova de conceito da utilização da mineração de texto. Os resultados obtidos, mostraram a viabilidade de uso das técnicas no caso prático. O desenvolvimento do aplicativo também demonstrou a possibilidade de desenvolvimento de aplicativos para uso direto pelo usuário, permitindo que a mineração de texto não fique restrita somente a especialistas.

Como sugestão de trabalhos futuros, propomos: a utilização de outros algoritmos, por exemplo, os baseados em redes neurais; a realização de ajustes mais refinados nos parâmetros dos algoritmos utilizados; a utilização de n-grams; um estudo mais detalhado dos termos extraídos do texto a fim de identificar novas stopwords; a utilização da linguagem python com o scikit learn [9].

References

[1] de Gestão e Desenvolvimento Institucional. VI Plano Diretor da Embrapa 2014-2034, 2015.
[2] THE TEXT MINING HANDBOOK: Advanced Approaches in Analyzing Unstructured Data - The Text Mining HandBook.pdf.
[3] Patrícia Maia, Rommel N. Carvalho, Marcelo Ladeira, Henrique Rocha, and Gilson Mendes. Application of text mining techniques for classification of documents: a study of automation of complaints screening in a Brazilian Federal Agency.
[4] Gustavo CG van Erven, Rommel N. Carvalho, Maristela Holanda, Marcelo Ladeira, Henrique Rocha, and Gilson Mendes. "Who is their mother?": A classification work to get answers over registration people databases.
[5] André L. Floriano and Celso AA Kaestner. Experimentos de Mineração de Dados em uma Base de Dados Utilizando Linguagem R.
[6] Pete Chapman, Julian Clinton, Randy Kerber, Thomas Khabaza, Thomas Reinartz, Colin Shearer, and Rudiger Wirth. CRISP-DM 1.0 Step-by-step data mining guide. 2000.

[8]https://github.com/cmacedo80/mestradocfmmdm
[9]http://scikit-learn.org/