3
Algoritmos propostos

Nesse trabalho foram desenvolvidos dois algoritmos que permitem classificar documentos em categorias de forma automática, com treinamento feito por usuários.

Tais algoritmos podem ser integrados em um ambiente completo de mineração de textos composto dos seguintes passos:

1. O usuário informa um conjunto de categorias de interesse e um conjunto de fontes de notícias na web. Por exemplo: categorias como soja, juros e violência das fontes de notícia Reuters, o Globo.
2. O programa retorna um conjunto de notícias das respectivas fontes especificadas pelo usuário.
3. O usuário classifica as notícias nas categorias anteriormente especificadas, tendo a possibilidade de classificar uma notícia em mais de uma categoria.
4. O programa retorna um novo conjunto de notícias classificadas nas categorias, podendo uma notícia estar classificada em mais de uma categoria.
5. O usuário avalia a classificação executada pelo programa e pode corrigir os falsos positivos e falsos negativos.
6. Volta para o passo 4.

A fase de treinamento e a fase de classificação ocorrem em conjunto com a realimentação do usuário, diferente do método clássico de treinamento em que primeiramente o algoritmo é treinado com os documentos para posterior classificação dos novos documentos nas categorias especificadas.

Nesse trabalho, são propostos dois algoritmos para resolver o problema da classificação automática de textos. Os algoritmos propostos são baseados no

método de classificação automática de textos naive Bayes utilizando o modelo multinomial e amortização de Laplace, descritos na seção 2.8.

3.1 Classificador pseudo-multirótulo

O primeiro algoritmo, chamado de pseudo-multirótulo, transforma o problema multirótulo em um problema unirótulo. Nesse algoritmo, o conjunto de categorias é estendido para suportar as combinações de categorias. Portanto, uma combinação de categorias também é considerada uma categoria.

Considere um conjunto de categorias $C = \{c_1, c_2,..., c_{|C|}\}$ e um conjunto de documentos $D = \{d_1, d_2,..., d_{|D|}\}$. Cada documento d_i está associado a uma combinação das categorias em C, que pode ser representado por um vetor binário $\vec{l} \in \{0,1\}^{|C|}$, onde a posição i está associada à categoria c_i, um valor 1 na posição i representa que o documento está associado à categoria c_i e um valor 0 representa que o documento não está associado à categoria c_i.

O conjunto de rótulos do algoritmo é definido como um conjunto $L \subseteq \{0,1\}^{|C|}$ de vetores binários $\vec{l}_j$, onde cada rótulo define um conjunto de documentos $\{d \in D \mid \forall i(\Phi(d, c_i) = \vec{l}[i])\}$. Por exemplo, o rótulo (0, 1, 1) representa o conjunto de documentos associados exatamente às categorias c_2 e c_3. Uma representação mais intuitiva do rótulo (0, 1, 1) seria $(\overline{c_1}, c_2, c_3)$.

Na fase de treinamento, dado um documento d_i associado ao rótulo $\vec{l}_j$, verifica-se se já foi realizado algum treinamento com o rótulo $\vec{l}_j$. Caso não tenha ocorrido, é criada uma nova categoria representada pelo rótulo $\vec{l}_j$. As palavras do documento d_i e suas freqüências são incluídas na categoria.

Já na fase de classificação, dado um documento d_i não conhecido, deve-se calcular a probabilidade $P(\vec{l}_j \mid d_i)$ do documento pertencer a cada uma das combinações de categorias representadas pelos vetores binários em $\vec{l} \in \{0,1\}^{|C|}$ e escolher a combinação com maior probabilidade. Adaptando-se a fórmula (22) para contemplar combinações de categorias, representadas por vetores, calcula-se:

$$\vec{l}^{*}(d_i) = \arg\max_{\vec{l}_j} \left(\log P(\vec{l}_j) + \sum_{k=1}^{|V|} w_{ki} \log P(t_k \mid \vec{l}_j) \right) \quad (31)$$

Como exemplo da fase de treinamento, considere o conjunto de treinamento composto pelos documentos $\{d_1, d_2, d_3, d_4\}$ pelo conjunto de categorias simples $\{c_1, c_2, c_3\}$ e pelas associações $(d_1, \{c_1, c_2\})$, $(d_2, \{c_2\})$, $(d_3, \{c_2\})$ e $(d_4, \{c_1, c_2, c_3\})$.

Para o documento d_1, é verificado se já foi realizado algum treinamento com a categoria representada pelo rótulo (1, 1, 0) ou $(c_1, c_2, \overline{c}_3)$. Como não foi realizado nenhum treinamento, é criada uma categoria representada pelo rótulo $(c_1, c_2, \overline{c}_3)$. Essa categoria é composta por todos os documentos que estão associados à exatamente c_1 e à c_2. Após a criação da categoria, são retiradas as "stopwords" de d_1, todas as palavras são transformadas em minúsculas e são calculadas as freqüências de cada palavra no texto de d_1. Finalmente, as palavras e suas respectivas freqüências são incluídas na categoria $(c_1, c_2, \overline{c}_3)$.

No treinamento a partir do documento d_2, é criada uma categoria representada pelo rótulo $(\overline{c}_1, c_2, \overline{c}_3)$. Essa categoria é composta por todos os documentos que estão associados à exatamente c_2. Então, as palavras de d_2 e suas respectivas freqüências são incluídas na categoria $(\overline{c}_1, c_2, \overline{c}_3)$.

Já no treinamento a partir do documento d_3 não é criada nenhuma categoria, pois a categoria associada ao documento d_3 já foi criada anteriormente.

Na fase de classificação, dado um documento d_5 não conhecido pelo sistema, o algoritmo calcula $\vec{l}^*(d_5)$, onde $\vec{l}_j \in \{(c_1, c_2, \overline{c}_3), (\overline{c}_1, c_2, \overline{c}_3), (c_1, c_2, c_3)\}$.

O problema de tal técnica é que são necessários exemplos de documentos para cada combinação de categorias, o que nem sempre é possível.

Além disso, um documento exemplo de uma combinação de n categorias (por exemplo, um documento pertencente à combinação "milho, grão, agricultura") não seria também tratado como exemplo das 2^n-2 subcategorias (por exemplo, "grão e agricultura", "milho e agricultura", "milho e grão", "milho", "grão", "agricultura").

A complexidade teórica da fase de treinamento do algoritmo é $O(|T_r||V|)$. Desconsiderando a fase de contagem da freqüência das palavras em cada documento, para cada documento do conjunto T_r, são incluídas todas as suas palavras na combinação de categorias associada, o que custa no pior caso, $O(|V|)$.

Já a fase de classificação tem a complexidade teórica de O ($|T_e||V||T_r|$). Para cada documento, é necessário selecionar a combinação de categoria com maior probabilidade do conjunto L de combinações de categorias criadas na fase de treinamento. Para selecionar a combinação de categoria com maior probabilidade se gasta O($|V||T_r|$), uma vez que, no pior caso, existem $|T_r|$ combinações de categorias, ou seja, um documento de treinamento para cada combinação.

3.2 Classificador multirótulo

Nesse algoritmo, um rótulo é uma combinação de categorias em C e é representado por um vetor $\vec{l}$, onde cada posição corresponde a uma categoria e pode assumir os valores: "0", "1" e "?". Um valor "1" na posição i do vetor $\vec{l}$ representa que o rótulo é composto de documentos associados à categoria c_i, um valor "0" representa a ausência de associações entre os documentos do rótulo e a categoria c_i, e um valor "?" representa que o rótulo pode ser composto tanto por documentos associados à c_i quanto a documentos não associados à c_i.

Formalmente, o rótulo $\vec{l}$ define um conjunto de documentos $\{d \in D \mid \forall i(\vec{l}[i] \neq ?) \rightarrow \vec{l}[i] = \Phi(d, c_i)\}$. Como exemplo, considere o rótulo (?, 1, 1). Tal rótulo define o conjunto de documentos que estão associados pelo menos às categorias c_2 e c_3. Uma forma mais intuitiva de representar o rótulo seria (c_2, c_3).

Na fase de treinamento, dado um conjunto D de documentos de treino, e um conjunto de categorias C, o algoritmo primeiramente cria dois conjuntos de rótulos $L \subseteq \{0,1\}^{|C|}$ e $L' \subseteq \{1,?\}^{|C|}$.

Para isso, para cada documento, é verificada a combinação de rótulos associada. Dado um documento d_i de treino e uma combinação de rótulos associada representada pelo vetor $\vec{l}_j$, é verificado se já foi criada uma categoria representada pelo vetor $\vec{l}_j$. Caso não tenha sido criada, são criadas duas categorias $\vec{l}_j \in \{0,1\}^{|C|}$ e $\vec{l}'_j \in \{1,?\}^{|C|}$, onde $\vec{l}_j$ é a categoria composta de documentos associados exatamente à combinação de rótulos de d_i e $\vec{l}'_j$ é a categoria composta de documentos associados pelo menos à combinação de rótulos de d_i.

Uma vez criadas todas as categorias, para cada documento d_i são calculadas as freqüências de cada palavra no texto de d_i. Verifica-se, então, a combinação $\vec{l}_j$ associada ao documento d_i e são incluídas as palavras do documento e suas freqüências na combinação de categorias representada pelo rótulo $\vec{l}_j$ e em todas as subcombinações $\vec{l}'_k \in L'$ onde $\forall i(\vec{l}'_k[i]=1) \rightarrow (\vec{l}_j[i]=1)$, ou seja, a combinação de categorias $\vec{l}'_k$ está contida na combinação $\vec{l}_j$. Por exemplo, se o documento d_i está associado ao rótulo $(c_1, c_2, \overline{c}_3, c_4)$, ele será incluído nos rótulos $(c_1, c_2, \overline{c}_3, c_4)$, (c_1, c_2, c_4), (c_1, c_2), (c_1, c_4), (c_2, c_4), (c_1), (c_2), (c_4).

Na fase de classificação, considere os conjuntos de rótulos criados pelo treinamento $L = \{\vec{l}_1, \vec{l}_2, ..., \vec{l}_{|L|}\} \in \{0,1\}^{|C|}$, $L' = \{\vec{l}'_1, \vec{l}'_2, ..., \vec{l}'_{|L'|}\} \in \{1,?\}^{|C|}$ e um documento d_i não conhecido pelo classificador.

Primeiramente, deve-se calcular a probabilidade $P(\vec{l}'_j \mid d_i)$ do documento pertencer a cada uma das combinações $\vec{l}'_j \in L'$ compostas apenas de uma categoria e escolher a combinação com maior probabilidade, representada por $\vec{l}'^{(1)}$.

Porém, não é necessário calcular exatamente $P(\vec{l}_j \mid d_i)$, uma vez que $P(d_i)$ é constante. Assim, através da equação (31), é escolhida a combinação $\vec{l}'^{(1)}$, que possui a maior probabilidade.

Posteriormente, escolhe-se a combinação com maior probabilidade $\vec{l}'^{(2)}$, do conjunto de combinações $\vec{l}'_j \in L'$ compostas por duas categorias, onde uma das categorias da combinação pertence à combinação $\vec{l}'^{(1)}$, escolhida anteriormente.

Então, se compara a probabilidade $P(\vec{l}'^{(2)} \mid d_i)$ com a probabilidade $P(\vec{l}^{(1)} \mid d_i)$, onde $\vec{l}^{(1)} \in L$ é a combinação tal que $\forall k(\vec{l}'^{(1)}[k]=1) \rightarrow (\vec{l}^{(1)}[k]=1)$ e $\forall k(\vec{l}'^{(1)}[k]=?) \rightarrow (\vec{l}^{(1)}[k]=0)$. Por exemplo, se $C = \{c_1, c_2, c_3\}$ e $\vec{l}'^{(1)} = (c_1)$, então $\vec{l}^{(1)} = (c_1, \overline{c}_2, \overline{c}_3)$.

Para ser realizada a comparação das probabilidades, calcula-se $\arg\max_{\vec{r}} \left\{ P(\vec{r} \mid d_i), \vec{r} \in \{\vec{l}'^{(2)}, \vec{l}^{(1)}\} \right\}$. Se $P(\vec{l}'^{(2)} \mid d_i) < P(\vec{l}^{(1)} \mid d_i)$, então o documento d_i é classificado em $\vec{l}^{(1)}$. Caso contrário, o algoritmo continua aumentando a quantidade de categorias em cada combinação, até que a condição de parada

$P(\vec{l}'^{(k)} \mid d_i) < P(\vec{l}^{(k-1)} \mid d_i)$ seja satisfeita, ou a quantidade de categorias em cada combinação seja |*C*|. Caso a quantidade de categorias em cada combinação seja |*C*| e a condição de parada não seja satisfeita o documento d_i é classificado na combinação de categorias $\vec{l}^{(k)}$, que representa a combinação composta por todas as categorias em *C*.

Como exemplo, considere um documento d_1, um conjunto *C* de categorias $\{c_1, c_2, c_3\}$ e um conjunto de rótulos $L' = \{(c_1), (c_2), (c_1, c_2), (c_1, c_3)\}$. Inicialmente, o algoritmo induz uma associação entre o documento d_1 e uma das categorias em *C*, calculando a probabilidade do documento d_1 pertencer a cada uma das categorias em *C*. Caso a categoria induzida tenha sido c_1, são analisadas todas as combinações de categorias presentes no conjunto de rótulos *L´*, onde uma das categorias é c_1, ou seja, (c_1, c_2) e (c_1, c_3). Então, é escolhido o rótulo com maior probabilidade. Caso o rótulo com maior probabilidade seja (c_1, c_2), a regra de decisão deverá induzir se o documento pertence exatamente a c_1 ou se pertence pelo menos a c_1 e c_2. A regra de decisão pode ser visualizada pela figura 2. O algoritmo induz se o documento pertence à região marcada na figura com linhas diagonais, da região marcada com linhas horizontais.

$$P(c_1, c_2 \mid d_1) > P(c_1, \overline{c}_2, \overline{c}_3 \mid d_1)?$$

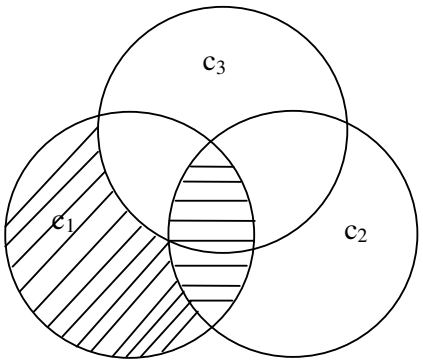

Figura 2 - Exemplo da regra de decisão do algoritmo multilrótulo proposto.

A complexidade teórica da fase de treinamento do algoritmo é $O(|T_r||V||T_e|)$. Para cada documento pertencente ao conjunto de treinamento T_r, no pior caso, existem T_r combinações de categorias pertencentes ao conjunto de combinações de categorias.

A complexidade teórica da fase de classificação do algoritmo é $O(|T_r||V||T_e|)$. No primeiro passo, é selecionada a combinação de categorias composta de apenas 1 categoria. Uma vez escolhida a combinação de categorias composta de apenas 1 categoria, seleciona-se a combinação de categorias compostas de 2 categorias, onde uma das categorias pertence à combinação de categorias escolhida no primeiro passo. Assim, no pior caso, compara-se todas as combinações de categorias existentes na base de treinamento.

Como exemplo da fase de treinamento, considere o conjunto de treinamento composto pelo conjunto de documentos $D = \{d_1, d_2, d_3, d_4, d_5\}$, pelo conjunto de categorias $C = \{c_1, c_2, c_3, c_4\}$ e pelas associações $(d_1, \{c_1, c_2\})$, $(d_2, \{c_2\})$, $(d_3, \{c_1, c_2, c_3\})$, $(d_4, \{c_1, c_3\})$ e $(d_5, \{c_1\})$.

Primeiramente, são criados os conjuntos de rótulos L e L'. Para isso, são verificados para cada documento as respectivas categorias associadas. Assim, para o documento d_1, são criados os rótulos (c_1, c_2) e $(c_1, c_2, \overline{c}_3, \overline{c}_4)$, para d_2, (c_2) e $(\overline{c}_1, c_2, \overline{c}_3, \overline{c}_4)$, para d_3, (c_1, c_2, c_3) e $(c_1, c_2, c_3, \overline{c}_4)$, para d_4, (c_1, c_3) e $(c_1, \overline{c}_2, c_3, \overline{c}_4)$ e d_5, (c_1) e $(c_1, \overline{c}_2, \overline{c}_3, \overline{c}_4)$.

Desta forma os conjuntos de rótulos são $L' = \{(c_1), (c_2), (c_1, c_2), (c_1, c_3), (c_1, c_2, c_3)\}$ e $L = \{(c_1, \overline{c}_2, \overline{c}_3, \overline{c}_4), (\overline{c}_1, c_2, \overline{c}_3, \overline{c}_4), (c_1, c_2, \overline{c}_3, \overline{c}_4), (c_1, \overline{c}_2, c_3, \overline{c}_4), (c_1, c_2, c_3, \overline{c}_4)\}$.

Depois de criados os conjuntos de rótulos, para cada documento são analisadas suas palavras, são retiradas as "stopwords", calculadas suas freqüências no texto e inseridas nos rótulos ao qual o documento pertence.

Para o documento d_1, suas palavras e freqüências são incluídas nos rótulos $(c_1, c_2, \overline{c}_3, \overline{c}_4)$, (c_1, c_2), (c_1) e (c_2).

Para o documento d_2, suas palavras e freqüências são incluídas nos rótulos $(\overline{c}_1, c_2, \overline{c}_3, \overline{c}_4)$ e (c_2).

Para o documento d_3, suas palavras e freqüências são incluídas nos rótulos $(c_1, c_2, c_3, \overline{c}_4)$, (c_1, c_2, c_3), (c_1, c_2), (c_1, c_3) e (c_2).

Já no caso do documento d_4, suas palavras e freqüências são incluídas nos rótulos $(c_1, \overline{c}_2, c_3, \overline{c}_4)$, (c_1, c_3) e (c_1).

Finalmente para o documento d_5, suas palavras são incluídas nos rótulos $(c_1, \overline{c}_2, \overline{c}_3, \overline{c}_4)$ e (c_1).

Como exemplo de classificação, considere o documento d_6 não conhecido pelo classificador.

Primeiramente, o classificador escolhe o rótulo com uma categoria com maior probabilidade. Assim, o classificador deve calcular $\arg\max_{\vec{r}} \left\{P(\vec{r} \mid d_i), \vec{r} \in \{(c_1),(c_2)\}\right\}$.

Supondo que o rótulo escolhido tenha sido (c_1), então o classificador analisa todos os rótulos com duas categorias, onde uma das categorias é c_1, ou seja, os rótulos (c_1, c_2) e (c_1, c_3). Caso o rótulo com maior probabilidade seja (c_1, c_3), então o classificador compara $P(c_1, c_3 \mid d_6)$ com $P(c_1, \overline{c}_2, \overline{c}_3, \overline{c}_4 \mid d_6)$. Se $P(c_1, c_3 \mid d_6) < P(c_1, \overline{c}_2, \overline{c}_3, \overline{c}_4 \mid d_6)$, o documento d_6 é classificado no rótulo $(c_1, \overline{c}_2, \overline{c}_3, \overline{c}_4)$.