Análise de Classificadores de Questões

André Andrade e Paulo Sérgio

Recife, 2019

1. INTRODUÇÃO

Neste artigo, avaliamos a eficácia de algoritmos de aprendizagem de máquina na classificação de perguntas de acordo com seus assuntos. A importância disso surge quando um usuário tenta recuperar informações em documentos, pois, dada a potencial grandiosidade da tarefa e o tédio de ter de analisar inúmeros possíveis documentos, delegar essa atividade a um algoritmo de inteligência artificial acaba sendo a melhor alternativa. Além disso, esse tipo de algoritmo também é de grande valor na criação de chatbots, para que possam identificar sobre o que está sendo perguntado por um usuário.

No TREC (Text Retrieval Conference), um número de perguntas e respostas tentam responder uma lista de perguntas predefinidas a partir do uso de um conjunto de documentos predeterminado. Nosso objetivo é encontrar qual é o método mais preciso na predição sobre o que as perguntas estão falando conforme suas categorias.

2. MATERIAIS

Utilizamos a base de dados organizada por Xin Li e Dan Roth, que foi utilizada em seu artigo Learning Question Classifiers (2002).

Usamos três tipos de pré-processamento para cada tipo de algoritmo escolhido. Serão eles o BOW (Bag of Words), o TF (Term Frequency) e o TF-IDF (Term Frequency – Inverse Document Frequency).

No teste em que foi realizado, foi utilizado o banco de dados https://cogcomp.seas.upenn.edu/Data/QA/QC/train_1000.label para fazer o treinamento e o banco de dados https://cogcomp.seas.upenn.edu/Data/QA/QC/TREC_10.label.

No nosso conjunto de treinamento, possuíamos 1000 perguntas com categorias e subcategorias, onde os algoritmos foram treinados para classifica-los baseado em sua devida categoria. Trata-se, assim, de um modelo de classificação mais ampla do que o usado por Xin Li e Dan Roth.

Possuíamos nesse conjunto de treinamento 6 categorias (['DESC', 'ENTY', 'A BBR', 'HUM', 'NUM', 'LOC']), das quais elas aparecem nas quantidades (211, 244, 18, 2 20, 151, 156) respectivamente, o que faz com que o conjunto seja desbalanceado.

3. MÉTODOS

Os algoritmos utilizados serão o KNN (K-Nearest Neighbors – modelo do vizinho mais próximo), o Naive Bayes e redes neurais com MLP (Multilayer Perceptrons).

3.1. KNN

Para classificarmos conforme o KNN, primeiro encontramos $f(k, x_q)$ e tomamos o voto da maioria dos vizinhos (que é o voto majoritário, em caso de classificação binária). Para evitar empates, k é sempre escolhido como número ímpar. Para fazer regressão, podemos tirar a média ou mediana de k vizinhos, ou podemos resolver um problema de regressão linear sobre os vizinhos (Russel e Norvig, 2004).

A função f é definida assim:

$$f(x_q) = \arg\max_{v \in V} \sum_{i=1}^{k} \delta w_i(v, f(x_i))$$

Onde:

x_q é uma instância;

w_i é o peso atribuído segundo a distância; e

$$\begin{cases} \delta\big(v, f(x_i)\big) = 1, \text{ se } v = f(x_i) \\ \delta\big(v, f(x_i)\big) = 0, \text{caso contrário} \end{cases}$$

3.2. Naive Bayes

O teorema de Bayes é utilizado para calcular a probabilidade de um evento ocorrer baseado no conhecimento (a priori) que pode estar relacionado a esse próprio evento. Nesse teorema, ele mostra como modificar as probabilidades a priori para obter probabilidades a posteriori.

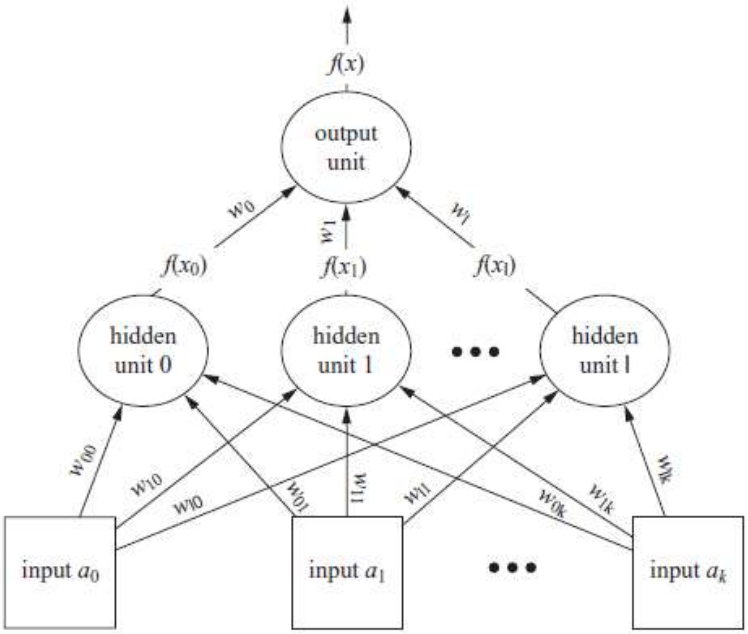
A definição formal é expressa matematicamente por essa equação:

$$P(A \mid B) = \frac{P(B \mid A)\, P(A)}{P(B)},$$

em que A e B são eventos e $P(B) \neq 0$.

3.3. MLP

MLP (Multilayer Perceptrons) são redes neurais compostas por várias camadas, podendo haver camadas escondidas. O esquema é dado de acordo como a figura abaixo:



MLP com uma camada escondida (Whitten, Frank e Hall, 2011).

As redes neurais são compostas por nós ou unidades conectadas por ligações direcionadas (Russel e Norvig, 2004). Olhando para a figura, é possível perceber que cada ligação tem um peso numérico associado a ele, que é o que determina a força do sinal de conexão.

A função de ativação que consideramos aqui é a *sigmoid*, dada por:

$$f(x) = \frac{1}{1+e^{-x}}$$

A função que calcula o erro mais usada é a que mede o erro quadrático médio. Assim, para uma única instância de treinamento:

$$E = \frac{1}{2}(y - f(x))^2$$

Por fim, para a suavização do erro, aplica-se o gradiente descendente conforme a equação:

$$\frac{dE}{dw_{ij}} = (f(x) - y)f'(x)w_i f'(x_i)\, a_j$$

com relação a cada erro w_{ij}.

4. EXPERIMENTOS E RESULTADOS

4.1. KNN

Método de Pré-Processamento	k Ótimo	Acurácia
BOW	2	0,658
Tf	2	0,738
Tfidf	12	0,754

4.2. Naive Bayes

Método de Pré-Processamento	Acurácia
BOW	0,542
Tf	0,49
Tfidf	0,684

4.3. MLP

Método de Pré-Processamento	Número de Camadas Escondidas Ótimo	Acurácia
BOW	37	0,788
Tf	4	0,792
Tfidf	64	0,81

5. CONCLUSÃO

Baseado nos resultados em que obtivemos nos classificadores KNN, Naive Bayes e MLP, chegou- se a conclusão que para esse problema de classificação de questões a melhor opção foi o MLP, que em todos os resultados comparando com os devidos pré-processamentos dos outros algoritmos se manteve superior em todos os casos, seguido do KNN e por último o Naive Bayes, que apresentou um resultado não tão satisfatório, porém, comparado aos outros algoritmos obteve uma velocidade bem acima.

REFERÊNCIAS BIBLIOGRÁFICAS

RUSSEL, S.; NORVIG, P. **Inteligência Artificial** - Tradução da Segunda Edição. Rio de Janeiro: Elsevier, 2004.

WITTEN, I. H; FRANK, Eibe; HALL, Mark A. **Data mining**: *practical machine learning tools and techniques*. 3rd ed. Burlington, MA: Elsevier/Morgan Kaufmann, 2011.

LI, Xin; ROTH, Dan. **Learning Question Classifiers**. *COLING'02*, Agosto, 2002.

ITTYCHERIAH, Abraham; FRANZ, Martin; ROUKOS, Salim. **IBM's Statistical Question Answering System – TREC - 10**. 2001.