Análise de Sentimentos de comentários da base de dados IMDb

Guilherme Duarte[1] e Ana Pinto[1]

[1] Instituto Universitário de Lisboa, Cidade Universitária de Lisboa, Av. das Forças Armadas, 1649-026 Lisboa

Abstract. A análise de sentimentos é uma das tarefas mais importantes e difíceis numa organização. Consiste na atribuição de uma opinião, atitude ou sentimento (felicidade, raiva, ódio, …) a uma frase ou um texto de uma opinião em relação a um produto ou tópico. Neste trabalho começou-se por criar uma *baseline* para comparar com outros métodos testados. De seguida houve uma tentativa de melhorar os resultados de duas formas: através de um léxico de sentimentos e através de um pré-processamento e de um processamento da língua natural (NLP) com algoritmos de *Machine Learning*. A eliminação de carateres, dígitos e acentos e seleção do número de *features* demonstram ser os processos que conduzem a resultados de *accuracy* superiores. O classificador BernoulliNB demonstrou ser o mais vantajoso para os dados em estudo.

Keywords: NLP, Análise de Sentimentos, IMDb.

1 Introdução

A análise de sentimentos (AS) é uma das áreas de *text mining* que utiliza Processamento de Linguagem Natural (PLN) e consiste na extração e identificação do sentimento (positivo, negativo e em alguns casos neutro) que um utilizador/escritor expressa em relação a um objeto [1].

A análise de sentimentos tem sido aplicada para a deteção de sentimentos relacionados com os comentários que são feitos aos filmes existentes no site da Internet Movie Database (IMDb). O site inclui avaliações relacionadas com filmes, desde os autores até ao elenco. A análise destes comentários permite aferir qual a opinião geral que os utilizadores têm em relação aos filmes o que pode ser usado pela indústria cinematográfica de forma a ajustar o produto àquilo que o público quer. De realçar que os comentários publicados nos websites são geralmente informais e não estruturados gramaticamente pelo que requerem algum pré-processamento.

O objetivo deste trabalho é identificar qual a melhor metodologia para a identificação dos sentimentos relativos aos comentários do IMDb. Com esse fim, foram estudados vários tipos de pré-processamento e classificadores. Segue-se uma breve descrição das etapas envolvidas na tarefa de análise de sentimentos.

1.1 Pré-processamento

A etapa de pré-processamento é a que antecede a aplicação do algoritmo no conjunto de treino e visa a preparação e limpeza dos dados, de forma a que o algoritmo possa ser aplicado de forma mais eficiente. Esta etapa pode também anteceder a extração e seleção de *features*, infra descrita. Existem diversos métodos que podem ser, dependem do conjunto de dados que estamos a analisar, para o trabalho importa realçar os seguintes:

- Remoção de carateres estranhos, acentos e números: os comentários disponíveis *online* muitas vezes contêm carateres e números que devem ser removidos porque não contribuem para a análise de sentimentos;
- Remoção de *stopwords*: consiste na remoção de palavras que são comuns e não acrescentam informação sobre o sentimento da frase (ex. determinantes, preposições, etc.), permite diminuir o tamanho do documento que está a ser analisado;
- *Stemming*: permite a eliminação dos afixos, reduzindo as palavras ao seu *stem*. Não tem em conta se os verbos são regulares ou não, pelo que pode cortar as palavras de uma forma que não faz sentido;
- *Lemming*: permite a redução das palavras para o seu radical mas tem em conta a morfologia da mesma, ou seja, não corta as palavras de forma indiscriminada. Esta técnica é feita com base num dicionário que contem os diferentes lemas;
- Expansão das contrações: permite a alteração das contrações para a sua forma original, auxiliando na tarefa de tratamento da negação (ex. don't para do);
- Part-of-Speech (POS) *Tagging*: técnica que categoriza as palavras tendo em conta a classe gramatical a que pertencem, isto é, nomes, verbos, advérbios, etc. , tendo em conta a relação dessa palavra com as palavras adjacentes [2].

1.2 Extração e seleção das *features*

A extração das *features* corresponde à tarefa de transformar os dados numa representação numérica enquanto que a seleção das *features* tem como objetivo escolher as mais importantes, diminuindo assim a sua dimensionalidade e aumentando a performance do classificador. O número e qualidade das *features* compromete o desempenho do classificador; *features* pouco relevantes podem necessitar de um modelo mais complicado para atingir o mesmo desempenho que *features* corretamente escolhidas [3].

Existem diversas formas de extrair as *features*, uma dessas técnicas é o *word count vector* que contabiliza a quantidade de vezes que uma palavra existe num determinado documento [5].

1.3 Técnicas de classificação

Os classificadores podem ser divididos em generativos e discriminativos. Os generativos, como os Naive Bayes (NB), constroem um modelo que explica os dados. Dada uma observação, retornam a classe com maior probabilidade de ter criado a observação[1]. Os classificadores discriminativos, como a Logistic Regression, Support Vector

Machines (SVM), K-Nearest Neighbour (KNN) e árvores de decisão(Decison Tree), criam modelos que explicam quais as *features* mais úteis para distinguir as classes. Geralmente, os algoritmos discriminativos são mais precisos e por isso são os mais usados [1].

Os classificadores lineares mais utilizados são as redes neuronais e os SVM. Um exemplo de um algoritmo de redes neuronais é o Multilayer Perceptron (MLP)[6]. Os classificadores do tipo SVM constroem hiperplanos num espaço multidimensional para separar amostras de duas classes e são independentes do número de *features* [7].

Quanto aos classificadores probabilísticos, os mais utilizados são os NB e Logistic Regression (LR). Os classificadores NB consideram que as *features* são independentes e calculam a probabilidade de um comentário pertencer a uma determinada classe (positivo/negativo) através do teorema de Bayes [7]. Uma vantagem destes classificadores é que necessita de um conjunto de treino pequeno para o cálculo das métricas [4]. Por outro lado, os classificadores de Logistic Regression, não têm em conta a independência das *features*.

1.4 Medidas de validação

A *accuracy* é a medida mais intuitiva para avaliar os classificadorses e é dada pelo rácio das observações previstas em relação ao total de observações. Para conjuntos de dados não balanceados, não se deve usar a *accuracy* como a métrica principal. Contudo, tendo em vista que o nosso conjunto de dados é balanceado, esta foi a métrica escolhida para comparar os diferentes modelos [3].

2 Análise do Trabalho relacionado

Diferentes abordagens de análise de sentimentos têm sido descritas, algumas focam-se no pré-processamento enquanto outras realçam a importância da escolha dos classificadores. Segue-se uma breve revisão de 4 artigos que têm por base a análise a comentários do IMDb, como no presente trabalho.

Tripathy *et al.*, comparou a *accuracy* obtida com os classificadores MultinomialNB e SVM. A etapa de pré-processamento consistiu na eliminação dos carateres especiais, números, espaços em branco, carateres repetidos nas palavras e *stopwords*. Em seguida, extraiu as *features* e procedeu à sua vectorização através do TF e do TF-IDF. O maior valor de *accuracy* foi obtido com o classificador SVM [4]. À sua semelhança, um grupo da universidade de BRAC, estudou a influência de dois classificadores Naive Bayes, dois classificadores do tipo SVM e um classificador MLP. Foram utilizados 2 classificadores SVM, SMO com kernel poly e SMO com kernel RBF. O classificador kernel RBF demonstrou ser o melhor para a tarefa de análise de sentimentos destes dados. O impacto das *stopwords* foi estudado com o classificador MultinomialNB, tendo-se verificado que o uso de *stopwords* é benéfico para a previsão correta do comentário [1].

3 Descrição dos dados utilizados

O conjunto de dados foi extraído do site do IMDb correspondendo a 2000 *reviews* de filmes em inglês anotadas com as etiquetas "pos" e "neg", encontrava-se balanceado e previamente separado em conjunto de treino e teste (IMDB-sample). O conjunto de treino continha 49,4% de comentários negativos enquanto o conjunto de teste tinha 52,5%.

4 Descrição das tarefas implementadas

O trabalho foi dividido em 3 partes: Numa primeira parte foi criada uma *baseline*, a qual não sofreu qualquer tratamento, e que tal como o nome indica serviu de comparação (de base) para os métodos consequentes. Em todos os casos foi criado um bag-of-words model (BOW).

A. *Baseline*

A *baseline* foi classificada através de MultinomialNB(MNB) e BernoulliNB(BNB) com os valores que se encontram por padrão no site do scikit-learn e avaliada através da *accuracy* e nº de sentimentos atribuídos incorretamente.

B. Léxico de sentimentos

Na 2ª parte foi utilizado um léxico de sentimentos (*NRC Emotion lexicon* or EmoLex) com e sem tratamento da negação. No caso em que não houve o tratamento da negação as palavras foram passadas para letra minúscula (A1). Relativamente ao caso com o tratamento da negação, em que as contrações foram removidas, foram feitas duas análises. Na primeira não houve qualquer pré-processamento (à exceção da tokenização e passagem par minúscula) (A2) e no segundo caso foram removidas as *stopwords*, caracteres especiais (!, ?, …), acentos e dígitos (A3). Em todos os casos foi medida a *accuracy*, na **Fig. 1** encontra-se uma representação esquemática do processo seguido.

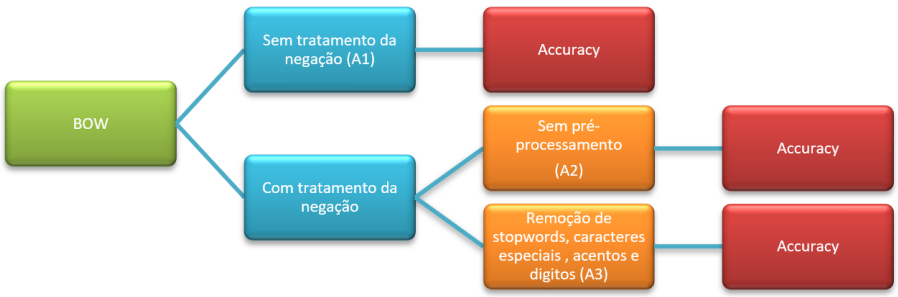

***Fig. 1** Processo utilizado para usar o léxico de sentimentos.*

C. *Machine Learning*

Na terceira parte foi feito aprendizagem automática sendo que inicialmente foram usadas várias técnicas de pré-processamento nomeadamente: *stemming com SnowBallstemmer* (B1), eliminação de acentos, caracteres especiais e dígitos (B2), *Lemming* (B3), remoção de *stopwords* (B4), seleção do número de *features* (B5) e POS tagging (B6) – esquematizado na **Fig. 2**.
Para a seleção das *features* mais relevantes foram testados 5 cenários que correspondem a 1000, 5000, 10000, 15000 e 20000 *features*. O número de 15000 *features* foi selecionado tendo em conta que foi para o qual se verificou o maior valor de *accuracy*.
Todas estas técnicas foram classificadas com BNB, MNB, K-Neighbors, Decision Tree (DT), Random Forest (RF), LR, C-Support Vector Classification (C-SVC), ADABoost e MLP. Para a seleção do número de *features* e POS foram apenas usados o MNB e BNB.

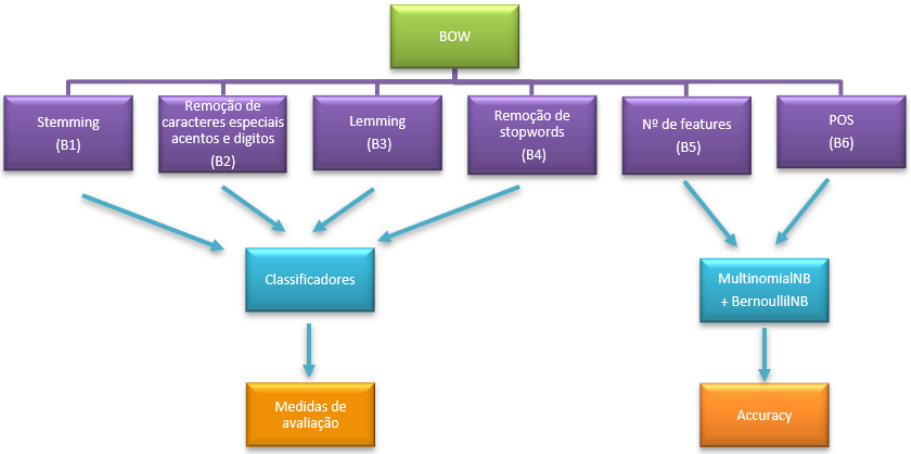

Fig. 2. *Processo utilizado na parte inicial da utilização de aprendizado de máquina.*

Por fim foram feitas várias combinações das técnicas anteriores (6 no total), como descrito na **Fig. 3**.

Nos cenários 5 e 6 foi feito GridSearch para otimizar os hiperparâmetros do modelo e *cross-validation* (10 *fold*) de forma a tornar o nosso modelo mais robusto. Nos 4 primeiros casos foram usados todos os modelos, enquanto que nos últimos dois foram usados os classificadores MultinomialNB e BernoulliNB (não otimizados), MLP, RF, C-SVC e LR.

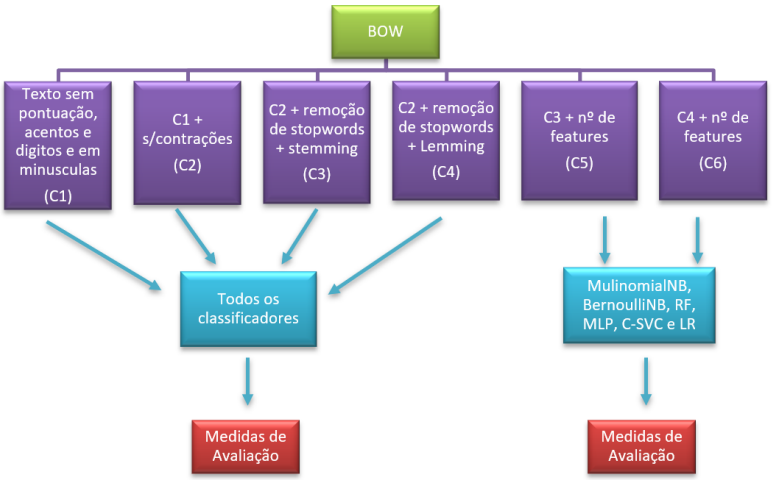

Fig. 3. *Processo utilizado na parte final da utilização de aprendizado de máquina.*

5 Resultados

Na **Tabela 1** é apresentado um resumo com os resultados obtidos para todos os classificadores, para as diferentes tarefas. Na **Fig. 4** é feita a comparação das *accuracys* obtidas para os melhores classificadores dentro de cada tarefa, para o conjunto de teste.

A partir dos dados obtidos podemos concluir que as tarefas B2, B5, C1 e C6 apresentaram melhores resultados relativamente à *baseline*. Estas tarefas estão relacionadas com um pré-processamento que implica a eliminação de carateres, dígitos e acentos e seleção do número de *features* ideal; pelo que a sua aplicação aparenta ser a que permite a criação de um modelo que melhor se ajusta aos dados em análise.

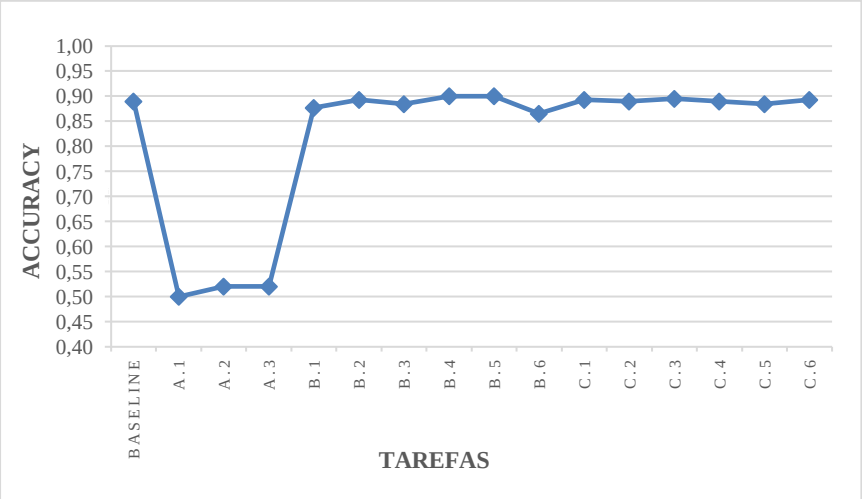

Fig. 4*. Comparação dos valores máximos de accuracy para as diferentes tarefas.*

É ainda possível observar que o uso do léxico de sentimentos degrada bastante o modelo relativamente à *baseline*, tendo-se verificado um decréscimo superior a 30%. Uma explicação para tal poderá dever-se ao *input* utilizado não ser o mais adequado. Podemos ainda verificar que os melhores modelos em cada um dos casos apresentam uma *accuracy* relativamente próxima, exceto os casos em que foi usado o léxico de sentimentos. Por fim, foi verificado que o modelo para o qual se obtém melhores resultados em termos de *accuracy*, foi o modelo em que foi feito a remoção das *stopwords* usando *Random Forest* e a seleção de 15000 *features* com o BernoulliNB.

O KNN destaca-se relativamente aos outros, produzindo resultados bastante fracos comparando com os outros modelos. Existem diversas razões para explicar os maus resultados para este caso, entre eles, a *accuracy* ter uma grande dependência da qualidade dos dados e sensibilidade à escala dos dados e a *features* irrelevantes, indicando que talvez as *features* utilizadas não são as mais adequadas [8].

Por fim podemos ver que BNB, RF, LR e DT foram os melhores classificadores consoante as tarefas implementadas, tendo o BNB sido o classificador em que mais tarefas obteve melhores resultados. Isto pode dever-se ao facto que cada *feature* é tratada de forma independente com valores binários apenas dando uma penalização ao modelo pela não ocorrência de nenhuma das *features* necessárias para prever o output, conseguir suportar bem *features* irrelevantes e de em pequenas quantidades de dados ou documentos conseguir dar resultados com maior *accuracy* e *precision* [9].

Tabela 1. *Resumos dos resultados obtidos com os diferentes classificadores para o conjunto de teste.*

Tarefas	*MNB*	*BNB*	*LR*	*SVC*	*KNN*	*DT*	*RF*	*MLP*	*AdaBoost*
Baseline	0,873	**0,890**	-	-	-	-	-	-	-
B1	0,873	0,878	0,855	0,808	0,625	0,840	0,875	0,845	0,860
B2	0,865	**0,893**	0,870	0,818	0,603	0,885	0,858	0,860	0,855
B3	0,878	0,885	0,873	0,805	0,593	0,875	0,878	0,875	0,878
B4	0,865	0,895	0,888	0,863	0,613	0,888	0,900	0,870	0,868
B5	0,878	**0,900**	-	-	-	-	-	-	-
B6	0,865	0,855	-	-	-	-	-	-	-
C1	0,865	**0,893**	0,870	0,818	0,603	0,888	0,868	0,855	0,855
C2	0,860	0,890	0,855	0,808	0,618	0,888	0,835	0,848	0,843
C3	0,858	0,880	0,885	0,840	0,578	0,895	0,890	0,853	0,850
C4	0,865	0,890	0,880	0,843	0,563	0,885	0,863	0,863	0,860
C5	0,863	0,883	0,885	0,870	-	-	0,845	0,863	-
C6	0,873	**0,893**	0,878	0,875	-	-	0,850	0,890	-

6 Conclusão e perspetivas futuras

A análise de sentimentos tem sido amplamente aplicada ao estudo das *reviews* do IMDb. Neste trabalho foram implementadas várias técnicas de *natural language processing* e *Machine Learning* com o intuito de identificar a melhor metodologia para a classificação de sentimentos das *reviews* mencionadas. Com base nos resultados obtidos, o uso do *NRC Emotion lexicon* não mostrou ser benéfico para a classificação de

sentimentos uma vez que levou à redução da *accuracy*. Em relação ao pré-processamento, os valores de *accuracy* obtidos foram bastante semelhantes para as diferentes tarefas, contudo pode destacar-se a tarefa da remoção de *stopwords* que resultou numa *accuracy* de 0,9 para o Random Forest e a seleção de 15000 features com o BernoulliNB.

Por fim, a combinação de pré-processamento e algoritmo que demonstrou melhores resultados foi a tarefa C3 com a aplicação do algoritmo de DecisionTree. O classificador BNB foi o classificador em que em mais tarefas obteve melhores resultados.

Tendo por base o trabalho supracitado é possível inferir sobre aspetos a melhorar/analisar em futuros trabalhos. Primeiramente, o uso de outro léxico de sentimentos (ex. SentiWordNet) e o tratamento da negação com base em bigramas e trigramas pode ajudar na obtenção de um modelo que se ajuste melhor aos dados. Outros tipos de pré-processamento podem ser tidos em conta por exemplo uso de *embeddings*; POS só com verbos, adjetivos ou nomes e *named entity tagger*. Por último, é relevante estudar a otimização da extração das *features* através do uso de diferentes técnicas (ex. Information Gain e Chi-Square).

Percentagem de contribuição para o trabalho: Ana 50% Guilherme 50%. Os dois mostraram ter compromisso com o trabalho, cumpriram os objetivos estabelecidos e trabalharam em equipa.

Referências

1. Jurafsky, D., Martin, J.H.: Speech and Language Processing 3nd edn, chapter 4. (2020).
2. Sahu , T.P, and Ahuja, S.: *Feature* Selection & Classification Algorithms. Conference: 2016 International Conference on Microelectronics, Computing and Communications (MicroCom) (2016)
3. Mtetwa, N., Awukam, A.O., Yousefi, M.: *Feature* extraction and classification of movie *reviews*. in 2018 5th International Conference on Soft Computing & Machine Intelligence (ISCMI). IEEE, pp. 67-71 (2019).
4. Tripathy, A., Agrawal, A., Rath, S. K.: Classification of sentimental *reviews* using *Machine Learning* techniques. Procedia Computer Science, vol. 57, pp. 821-829 (2015).
5. Birjali, M., Beni-hssane, A., Erritali, M.: Prediction of Suicidal Ideation in Twitter Data using *Machine Learning* algorithms. In International Arab Conference on Information Technology (ACIT) (2016).
6. Medhat, W., Hassan, A., Korashy,H.: Sentiment analysis algorithms and applications: A survey . Ain Shams Engineering Journal 5, 1093–1113 (2014).
7. Kumar, H. M. K., Harish, B. S., Darshan, H. K.: Sentiment Analysis on IMDb Movie *Reviews* Using Hybrid *Feature* Extraction Method. International Journal of Interactive Multimedia and Artificial Intelligence, Vol. 5, Nº 5 (2018).
8. Mutha, N.: https://www.mygreatlearning.com/blog/knn-algorithm-introduction/, last accessed 2021/04/16.
9. Chatterjee, M.: https://iq.opengenus.org/bernoulli-naive-bayes/, last accessed 2021/04/16.