# COM3110: Sentiment Analysis of Movie Reviews(2023/2024)
# Yi Li

## Introduction

Nowadays, rapid development of the Internet has brought a large amount of text data, which contains rich emotions and ideas. Sentiment analysis, as an important branch of natural language processing (NLP), has the main task of understanding and extracting emotional tendencies from text data, which has become a key tool for understanding public emotions and opinions. Especially in the film industry, audience reviews and feedback are crucial to the success of a film, affecting its box office. This project mainly uses multinomial naive Bayes and TF-IDF as the core algorithms of sentiment analysis.

## Implementation details

Bayes classifier: Navie_Bayes_model.py → 97 lines – 109 lines

$$s^* = \ argmax_{s_i} \frac{p(T|s_i)p(s_i)}{p(T)} \ = \ argmax_{s_i} p(T|s_i)p(s_i) \tag{1}$$

The goal of the classifier is to find a category s such that the probability $P(s|T)$ of this category is maximum given the text $T$. Due to evidence $P(T)$ is constant for all categories $s_i$ and we ignore it in our calculations.

Prior probability: Navie_Bayes_model.py → 36 lines – 57 lines

$$p(s_i) = \frac{count(s_i)+1}{\sum_{i-0}^{J} count(s_j)+ V} \tag{2}$$

In practice, the prior probabilities are calculated by simple relative frequencies, where $J$ is the number of different categories and $count(.)$ is the counting function. For smoothing, we make use of Laplace smoothing where $V$ stands for the amount of vocabulary.

Likelihood: Navie_Bayes_model.py → 36 lines – 57 lines

$$p(T|s_i) = p(t_1, t_2, \dots, t_N|s_i) \ \approx \ \prod_{j=1}^{N} p(t_j|s_i) \tag{3}$$

In the Naive Bayes classifier, we assume that all features are independent of each other. Therefore, the likelihood of judging text $T$ in category $s_i$ can be expressed as the product of the occurrence probabilities of all feature values $P(t|s)$

Feature extraction: NB_sentiment_analyser.py → 111 lines – 116 lines & utils.py 99 lines – 178 lines
We will discuss it in the 'Feature selection' session.

Macro-F1 score: NB_sentiment_analyser.py → 71 lines – 88 lines

## Feature selection

Inspired by Research by Zhaowei Qu et al., we also explored models incorporating TF-IDF weights. TF-IDF is a statistical method that evaluates the importance of a word to a document set or one of the documents in a corpus. In this method, the TF-IDF probability of each review is obtained by adding the TF-IDF values of the words that make up the sentence and dividing it by the total value of TF-IDF of different categories, which is then multiplied with the naive Bayes posterior probability as the judging criteria. We define them by the following formula:

$$all\_tfdf(\hat{s}) = \ \sum_{i=1}^{N} tfidf(\ s_i) \tag{4}$$

$$p_{tfidf}(T|\hat{s}_i) = \frac{p\left(\hat{t}_1, \hat{t}_2, \dots, \hat{t}_N \middle| \hat{s}_i\right)}{all\_tfidf(\hat{s}_i)} \approx \frac{\sum_{j=1}^{N} p(\hat{t}_j|\hat{s}_i)}{all\_tfidf(\hat{s}_i)} \tag{5}$$

$$s^* = argmax_{s_i} p(T|s_i)\, p(s_i)\, p_{tfidf}(T|\hat{s}_i) \qquad (6)$$

Where $all\_tfidf(.)$ calculates the corresponding TF-IDF value for each word. Then we sum the TF-IDF values of the words in each category separately. Next, we calculate the TFIDF probability $p_{tfidf}(T|s_i)$ for each sentence by adding the TF-IDF values of the sentence and dividing by the total TF-IDF value of each class respectively. Finally, the formula put into Naive Bayes to play a role in strengthening the features.

For our TF-IDF smoothing, we define that if there is a test word in the training set, the TF-IDF value in the training set is accumulated, otherwise it is accumulated by 1.

During the experiment on development dataset, we find that the Bayesian formula obtains the probability distribution of each class by taking chances. However, the results obtained by multiplying TFIDF value or using it alone as the basis for the probability distribution are greatly opposite, and the accuracy drops significantly. Thus, after repeated testing, we conclude that accumulating TF-IDF values and multiplying them with the Bayesian probability distribution gives the best results.

## Results and Discussion

We use the confusion matrix as well as macro-F1 to evaluate our model. The results show that there is a certain difference in the performance of the model when processing data sets with 5-Value and 3-Value.

| **Case** | **All_words** | | **Features** | |
|---|---|---|---|---|
| **Score** | Macro-F1 | Accuracy | Macro-F1 | Accuracy |
| **5-Value** | 0.348 | 0.376 | 0.344 | 0.355 |
| **3-Value** | 0.533 | 0.625 | 0.543 | 0.589 |

Table 1. Performance of Macro-F1 and Accuracy

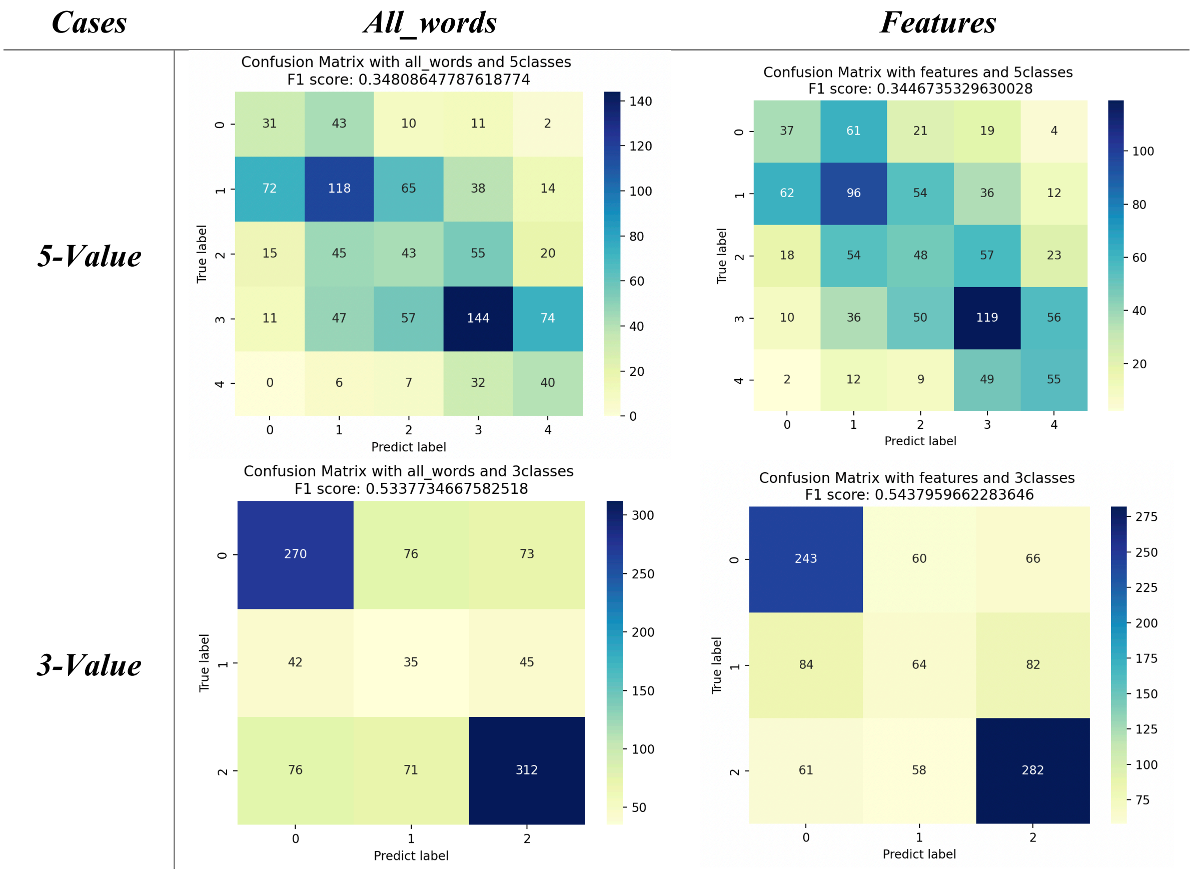

Table 2. Performance of 4 models with features and class value

As can be seen from the Table 1 and 2, the accuracy of our model in all words is generally higher than that of the model with features. In macro-F1, the score of the 3-value model with features is 1% higher than the non-features model, while the 5- The value model only reduces the macro-F1 score by 0.4%.

Overall, feature enhancement enables our model to identify more subtle differences in the data, thereby improving the model's delineation of boundaries between different categories. For the 5-value classification model, the macro-average F1 score is only reduced by 0.4% after adding features. Despite the slight decrease, this result still shows that the impact of feature enhancement on model performance is limited, especially when dealing with more granular emotion labels. For the 3-value case, the model seems to be better able to utilize features to distinguish emotional tendencies. This shows that for rough classification of emotions, our feature enhancement is more helpful to improve performance, while for more detailed emotion classification, more information other than features is needed to improve performance.

## Error analysis

| Class | Id | Review text | label |
|---|---|---|---|
| 3-Value | 1207 | Absurdities | 2 |
|  | 7971 | It irritates and saddens me that Martin Lawrence 's latest vehicle can explode obnoxiously into 2,500 screens while something of Bubba Ho-Tep 's clearly evident quality may end up languishing on a shelf somewhere . | 1 |
| 5-Value | 6400 | How about starting with a more original story instead of just slapping extreme humor and gross-out gags on top of the same old crap ? | 2 |
|  | 6336 | It 's a movie that ends with Truckzilla , for cryin ' out loud . | 1 |

Table 3. Examples that model have difficulty in classification

As discussed in ‘Results and Discussion’, our model is not good at processing complex long sentences or too short sentences, and we also found that for some interrogative sentences, judgment sentences, and non-emotional declarative sentences, the model will be significantly affected, resulting in some special Misjudgements such as nouns. Therefore, more detailed filtering of words is needed as a training set.

## Reference

“Improved Bayes Method Based on TF-IDF Feature and Grade Factor Feature for Chinese Information Classification | IEEE Conference Publication | IEEE Xplore,” ieeexplore.ieee.org. https://ieeexplore.ieee.org/abstract/document/8367204 (accessed Dec. 15, 2023).