Paper Template for COMP90049 Report

Anonymous

1 Introduction

Genre classification and sentiment analysis are the two most mainstream application scenarios of movie classification technology. The sentiment analysis focus on movie review comments to get objective audiences' attitudes. Genre classification technology is used not only to judge the movie ratings to find the suitable age group but also to define movie categories, such as action, comedy, horror and so on. It's a very challenging classification problem that labelling movies according to their corresponding genre.

In this paper, the research process can be mainly divided into two steps: i) feature selection and representation, ii) model training and prediction. The former relies on term frequency–inverse document frequency (TF-IDF), while the latter bases on the Naive Bayes and the Support Vector Machine (SVM). The report shows the changes in the prediction results under different features or different models and further analyzes these changes.

2 Related Work

Over the years, researchers have proposed various automatic genre classification methods for text and general video data. Some methods rely on textual features, such as movie titles or summary user tags. Other methods rely on audio-visual features.

Rasheed et al. compute four video features: lighting key, motion content, average shot length and colour variance to predict movie genres. But they only divided movies into four categories: comedy, action, drama, and horror. Obviously, this broad classification method is no longer applicable. In addition, since certain categories are very similar visually such as documentary and comedyusing just visual features are not enough. In this scenario, the audio or text extracted from audio is necessary for genre classification. One hybrid method was proposed by Huang and Wang combining both low-level visual features and audio information.

Ho applied three methods to predict 10 most popular genres based on the dataset only containing 16,000 movie titles. Overall, the SVM achieves the highest F1 score of 0.55 in his paper. Hoang explores Naive Bayes and Recurrent Neural Networks for text classification, and his model performs well in multi-label problems.

3 Feature Selection

3.1 Data Set

The data set used in this report is derived from a larger database script by Deldjoo et al. and F. Maxwell et al.

The data set is partitioned into three sets, a training set, a valid set and a test set. The training set and the valid set contain two files, the features data and the genres data. The training set is used to train and fit the model, and the valid set is used to test and analyze the differences between various models. The test set has only one file containing features data. We need to use a trained model to predict the genres of the test data. The prediction result is used for kaggle competition. Table 3.1 below shows the structure information of different files.

Corpus	Features	
Training set	train_features	5240 instances
	train_labels	5240 instances
Valid set	valid_features	299 instances
	valid_labels	299 instances
Test set	test_features	235 instances

Table 1: The structure of the data

3.2 Data Pre-processing

For the special model, the input data should have the same structure. Therefore, we should perform the same operation on train_features,

valid_features and test_features. Below I will only explain the processing of train_features.

Preprocessing the original data is necessary and effective to promote the training of the model, because there are NaN and invalid data in train_features. Abnormal data can be divided into three categories:

Since "MovieId" and "YTId" do not contain any information related to the movie category, eliminating them from the training set does not affect the accuracy of the model

The "year" represents the release time of the movie. But the data of "year" in some instances is missed. Compared with "title" and "tags", the "year" contains very little movie information, although movies of similar types may be released at the same time. I choose to exclude the "year" column.

In all instances, the values of "avf31", "avf32" and "avf104" are the same. I have reliable reasons to judge that these three data do not represent any characteristics of the movies. Therefore, I can eliminate these three columns.

3.3 Term Frequency–Inverse Document Frequency

Since the "title" and "tags" are textual data, We cannot directly know which word is important. Term Frequency–Inverse Document Frequency (TF-IDF) is a numerical statistic that is intended to reflect how important a word is to a document in the corpus.

Calculate the TF at first. The weight of a term that occurs in a document is simply proportional to the term frequency. In other words, in one instance, the more times a word appears, the more important the word. Therefore we can define:

$$tf(t,d) = f_{t,d}$$

$f_{t,d}$: the number of times that term t occurs in document d

Then, we need calculate the IDF. The specificity of a term can be quantified as an inverse function of the number of documents in which it occurs. In other words, A word is not important if it appears in most examples. So we can define:

$$idf(t,D) = log(D/n_t)$$

D: the number of instances n_t: the number of instances containing t.

Finally, we can get the tf-idf:

$$tf - idf = tf(t,d) * idf(t,D)$$

The greater the tf-idf, the more important the word.

3.4 Standardization or Normalization

Comparing different columns, we can find that the values between the columns are very different. For example, "avf1" is generally less than 0.5, but "avf15" is generally greater than 10. If we directly train this data, when "avf1" and "avf15" have the same coefficient, the impact of "avf15" will be much larger than that of "avf1". However, the data of different columns should have the same weight. Therefore, we need to standardize or normalize data.

By calculating the value of the Gaussian error function for each column, we can find that most data does not conform to the normal distribution. For example, "avf10" has only four discrete values. If we use standardization, the error of the training result may increase. Therefore, I use normalization to process the data, and the range of each column is [0,1].

$$X^{'} = (X - X_{min})/(X_{max} - X_{min})$$

4 Methodology

Because this is a classification problem, in a large number of machine learning models, we should first consider the classification algorithmsuch as Naive Bayes, SVM, K-nearest neighbours and Decision tree. I verified these four models in the code and found that Naive Bayes and SVM performed better. This article will focus on the in-depth analysis of Naive Bayes and SVM.

4.1 Naive Bayes

The naive Bayes method is based on the conditional independence assumption with the decision rule, *maximumaposteriori* (MAP), to classify an instance. It is a mainstream spam classification algorithm and has great advantages in processing textual data. In the original data, both the *title* and the *lag* are textual data, and naive Bayes may have an ideal training effect.

Naive Bayes can be roughly divided into three categories: Gaussian naive Bayes, Multinomial naive Bayes, and Bernoulli naive Bayes. When dealing with continuous data, one common assumption is the data is distributed according to a normal distribution. Therefore, we cannot calculate the exact probability of a giving point since the data is not discrete. However, we can get the mean and variance of the sample

to calculate the density function of normal distribution. Using the density function to replace the probability is the kernel of Gaussian naive Bayes.

When the data is discrete and finite, samples (feature vectors) represent the frequencies with which certain events have been generated by a multinomial. Multinomial naive Bayes is an event model commonly used for document classification, where events represent the occurrence of words in a single document.

Like the multinomial model, the Bernoulli model is also suitable for discrete features. The difference is that the value of each element in the Bernoulli model can only be 1 and 0. If we only need to judge whether the keyword appears, Bernoulli naive Bayes is the best model.

4.2 Support Vector Machine (SVM)

SVM is supervised learning models with associated learning algorithms. This method constructs a maximum-margin hyperplane which aims to lower the generalisation error of the classification. It can be used for both regression and classification tasks. Usually, it is widely used in classification labels.

With the help of kernel function, SVM can handle both linear and non-linear data classification. The most important theory is mapping the feature vectors in low-dimensional space to high-dimensional space where we can easiler to find linear separability. Besides, when we apply the kernel trick to maximum-margin hyperplanes, we can get a nonlinear classifier with curved borders.

5 Results

5.1 1

5.2 2

5.3 SVM

5.4 vsSVM

6 Conclusions

TTTTTTTT TTTTTTT