COMP30027 Machine Learning Report

Anonymous

1 Introduction

With the development and expansion of Internet, assorted kinds of blogs have been published by independent individuals to express their own ideas. Various information can be extracted from the contents of blogs, as raw data to analyse. Based on the several datasets provided (Schler et al., 2006), building a model by using supervised machine learning methods, which can be able to identify the age range of the author for each blog correctly, is required for this project.

In order to accomplish the task with high accuracy, a serious of feature engineering for enhancing the prediction, and evaluation among different implemented classifiers based on the performance, will be described in the following sections.

2 Feature Engineering

In this section, features as an important part of elements in prediction, will be considered for improving the performance of accuracy.

Figure 1 shows the distribution of top 30 words for age ranges, it can be noticed that only small amount of words has the ability to distinguish the age range. For example, "evermean" and "spanners" will be considered as categories of 34-36 age range, meanwhile, "jayel" and "levengals" will be categorized as 44-46 age range. On the other hand, there are still some balanced distribution in certain words which cannot be easily determined.

Moreover, from the instances given in the token training data, there are over 41% of instances are all labelled zero for 30 selected features. Furthermore, among the all zeros instances, the distributions of age classes are quite even, which would make the classifiers' prediction inaccurate and insufficient. In order to avoid all zero rows and increase the vary among instances, there are two strategies implemented correspondingly in subsection.

	14-16	24-26	34-36	44-46
anyways	0.653	0.323	0.022	0.002
cuz	0.748	0.214	0.036	0.001
digest	0.070	0.200	0.722	0.008
diva	0.051	0.205	0.744	0
evermean	0	0	1	0
fox	0.153	0.415	0.091	0.341
gonna	0.653	0.300	0.034	0.013
greg	0.145	0.444	0.107	0.304
haha	0.817	0.174	0.009	0.001
jayel	0	0	0	1
kinda	0.618	0.340	0.031	0.010
levengals	0	0	0	1
literacy	0.044	0.254	0.679	0.024
lol	0.778	0.144	0.038	0.041
melissa	0.324	0.357	0.027	0.292
nan	0.298	0.190	0.020	0.492
nat	0.407	0.163	0.004	0.426
postcount	0	0.478	0.522	0
ppl	0.855	0.136	0.009	0
rick	0.109	0.306	0.130	0.454
school	0.539	0.371	0.073	0.016
shep	0.015	0	0.010	0.974
sherry	0.044	0.190	0.051	0.714
spanners	0	0.01	0.99	0
teri	0.049	0.098	0.024	0.829
u	0.771	0.215	0.012	0.002
ur	0.841	0.155	0.004	0.000
urlink	0.193	0.691	0.174	0.028
wanna	0.701	0.268	0.026	0.006
work	0.227	0.618	0.127	0.028

Table 1: Distribution of Top 30 words

2.1 Feature Extraction and Selection

For the processing of re-extracting the token words, all raw texts from training data are stored for implementing the counting of both tokenization and occurrence. Among a huge number of extracted features, 1000 best features are selected by using χ^2 as score function parameter. Furthermore, all selected features with occurrences would be fitted and transformed into corresponding data representation for further modelling use. All implementations made for features are considered to increase the distinguishing in order to improve the prediction. Therefore, all raw data, including training, development and test, are processed the same way. Additionally, features which are not existed in either development or test data will be correspondingly count as zeros.

2.2 Data Pre-Processing

By the observation of raw training data, it can be noticed that groups of instance data are published by the same user, which can be considered from the same source. Therefore, grouping the instances by users is an appropriate and considerable data pre-processing step, in terms of increasing all the occurrences of instances data, which can improve model's the ability of distinguishing. Moreover, since groups of blogs are from the same user, the prediction of age range should be the same as well, which can avoid single instance predict differently than majority prediction from the same user.

3 Class Evaluation

In this section, evaluation of several classifiers over development data will be discussed below. Among various classifiers, three classifiers presenting high performances of accuracy are selected.
The accuracy calculated from development data is much lower than the accuracy by splitting the training data into training and test data, due to not only the model is overfitting to the training data, but also missing the instances of unlabelled class (presented as '?') in training data, which are unable to predict. Consider the age is between 14 and 46, but does not belong to any labelled classes, which would be quite hard to predict. Among all classifiers I implemented, unlabelled class is not handled in terms of not influencing the current prediction.

3.1 Decision Tree

Decision tree as a common method for prediction, which is easy to understand and interpret. With the default parameter, maximum depth as none, all of the nodes in tree will be expanded until all leaves are pure, which can distinguish the class no doubt. Nevertheless, the prediction made by decision tree classifier will be randomised at each split, which causes the differences of best split each time. Random state parameter can be used in order to maintain a certain result, which may not be the best. In this case, decision tree method is lack of consistency, so the accuracy fluctuates between 50% and 51%. To sum up, decision tree classifier cannot be able to guarantee the best split due to the random state, which is a limitation compared to random forest below.

	14-16	24-26	34-36	44-46	?
14-16	10114	2609	312	65	0
24-26	3357	11631	1894	416	0
34-36	362	1333	458	431	0
44-46	84	358	93	16	0
?	4997	5955	456	391	0

Table 2: Confusion Matrix (Decision Tree)

3.2 Random Forest with Grid Search

Basically, random forest method operates as a considerable amount of decision tree during the training time, which can be regarded as a better and more powerful version of decision tree method. Meanwhile, random forest can correct the overfitting produced by decision tree. Thus, the accuracy over the development data will be a step higher than the original result from decision tree method. Furthermore, since random forest classifier has a variety of parameters which can be adjusted in order to enlarge the accuracy. By using grid search, with various combination of parameters can be tested for selection, best parameters for use will be presented, generally number of trees in forest and features for best split, which can improve the performance of accuracy than the origin random forest method. From the confusion matrixes of both discussed models (see Table 2 and Table 3), it can be seen that more predictions are adjusted into classes which has more distribution, like 14-16 and 24-26. The concept of this classifier is to minimise the overall model variance, for gaining the higher accuracy.

	14-16	24-26	34-36	44-46	?
14-16	11212	1803	85	0	0
24-26	2149	14769	380	0	0
34-36	207	1957	420	0	0
44-46	38	409	104	0	0
?	4359	7043	397	0	0

Table 3: Confusion Matrix (Random Forest)

3.3 Logistic Regression

Compared to the linear regression, which can only analyse the continuous data, logistic regression is designed for analysing the categorical data, for example, the data representation created after the feature engineering. Instead of getting the certain category, comprehensive measured probability is calculated for each instance. Beyond that, logistic regression can discover the potential combined effects, which improve the prediction unconsciously. Therefore, logistic regression is a well-performed classifier with huge number of data and also suitable for frequency based features.

Although, various parameters are provided with the model. For example, random state sets as default none, which is randomised by random function in numpy array. After a serious of measurements, without using any additional parameters preforms the best result, even with random, it still predicts the same at last.

From the confusion matrix (see Table 4), all classes except 34-36 increase the correct predictions. Logistic regression model is the final submission with the highest accuracy, which is over 60% over the development data.

	14-16	24-26	34-36	44-46	?
14-16	12345	730	13	12	0
24-26	883	15304	1055	56	0
34-36	885	1414	275	10	0
44-46	17	181	275	78	0
?	4906	5638	1217	38	0

Table 4: Confusion Matrix (Logistic Regression)

4 Conclusions

14-16	98454
24-26	141104
34-36	30347
44-46	6510

Table 5: Age Distribution

Since the lack of instances between 34-36 and 44-46 in training data, most of distribution slants to the young generations, which make the classifier hard to predict these two age ranges, which is the reason why with the higher accuracy, the less predictions are made in this two classes.

Based on my own learning from the provided top token words, it can be discovered that almost all of the abbreviation words and internet slang are commonly used among the young generations, especially 14-16 age range. Some uncommon words can better distinguish a certain age range.

Moreover, compared among the predictions, (30 provided, 500 and 1000 features after processing of feature engineering and data pre-processing), all fitted into logistic regressions, it can be noticed that more features obtained, better predictions can make, which increases the accuracy significantly. Since the increasing number of features can make each instance unique to others, in order to gain the best suitable prediction of class.

In conclusion, both feature engineering and classifier selection can influence the final result of accuracy. In order to achieve higher performance, these two ways are necessary and complementary, which obtain over 65% on testing data.

References

Jonathan Schler, Moshe Koppel, Shlomo Argamon, and James Pennebaker. 2006. *Proceedings of 2006 AAAI Spring Symposium on Computational Approaches for Analyzing Weblogs.* Effects of Age and Gender on Blogging, Stanford, USA.