Project Machine Learning report for Milestone 3: The final prediction method

Zhanwang Chen

Abstract

This milestone is to evaluate the final prediction method that averaging the different models. And inspect the usefulness of the human and machine intelligence combination on question predicting.

1 Prediction methodology

Studies [1, 5] show that a combination of many different predictors can improve predictions. Two combination methods were experimented. Let X be the input space, f is the classifier, K is the classifier number.

mean combination rule[1]

$$f_j(x^1, \dots, x^R) = \frac{1}{R}\sum_{k=1}^{R} f_j^k(x^k). \tag{1}$$

product combination rule[1]

$$f_j(x^1, \dots, x^R) = \frac{\prod_{k=1}^{R} f_j^k(x^k)}{\sum_{j'}\prod_{k=1}^{R} f_{j'}^k(x^k)}. \tag{2}$$

Experiment results see appendix table 1, By combining the classifier with mean combination rule, can Indeed increase the robustness and the performance of the classification. For ngram features, the ensemble model got higher F1 score than every single classifier on both major and sub category, even the random forest model got very low F1 score but didn't hurt the overall ensemble model. As Figure 1 and Figure 2 shows. Ensemble model with ngram features got 9% improvement on major category, 13% improvement on sub category compare to average Macor-F1 score of the other single model (e.g svm and so on, ensemble model was not included). However, for product combination rule, the ensemble model tends to be affected and became deteriorated by bad performance classifier and didn't get significant improvement.

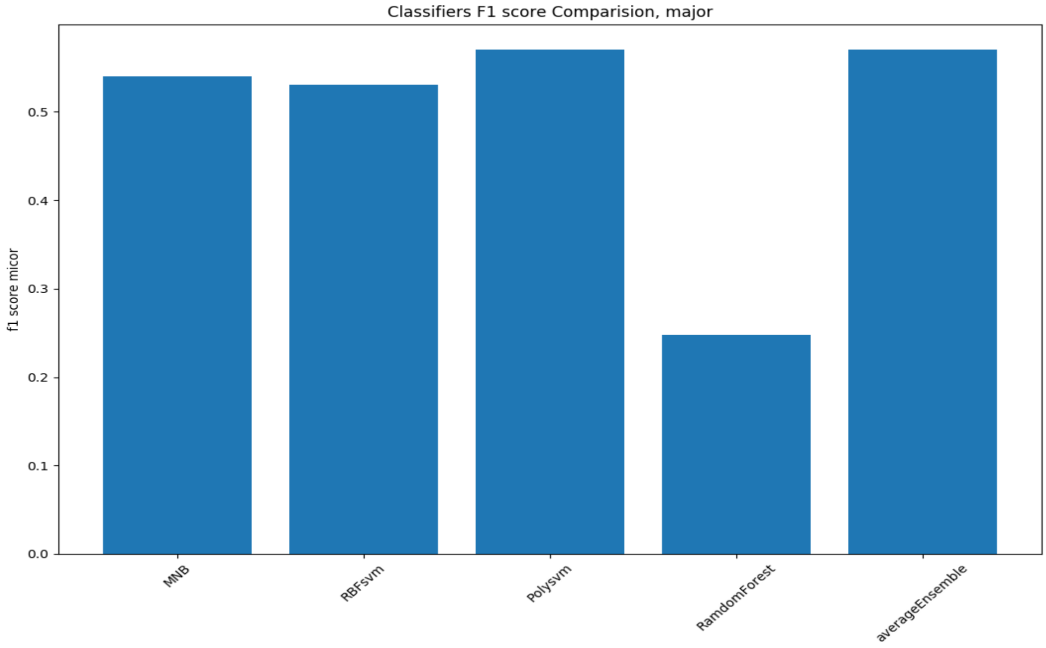


Figure 1: F1 score with mean average rule on major category

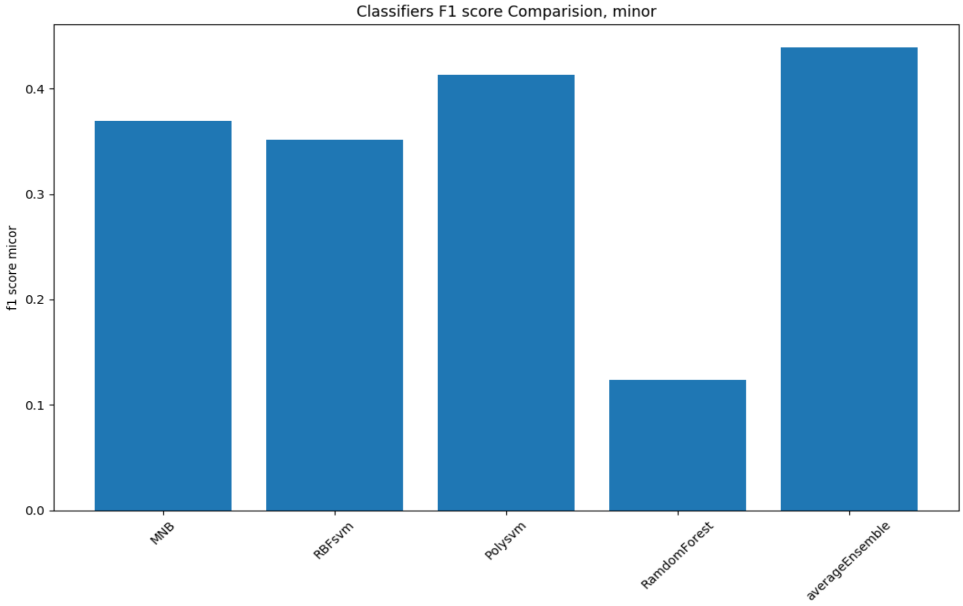


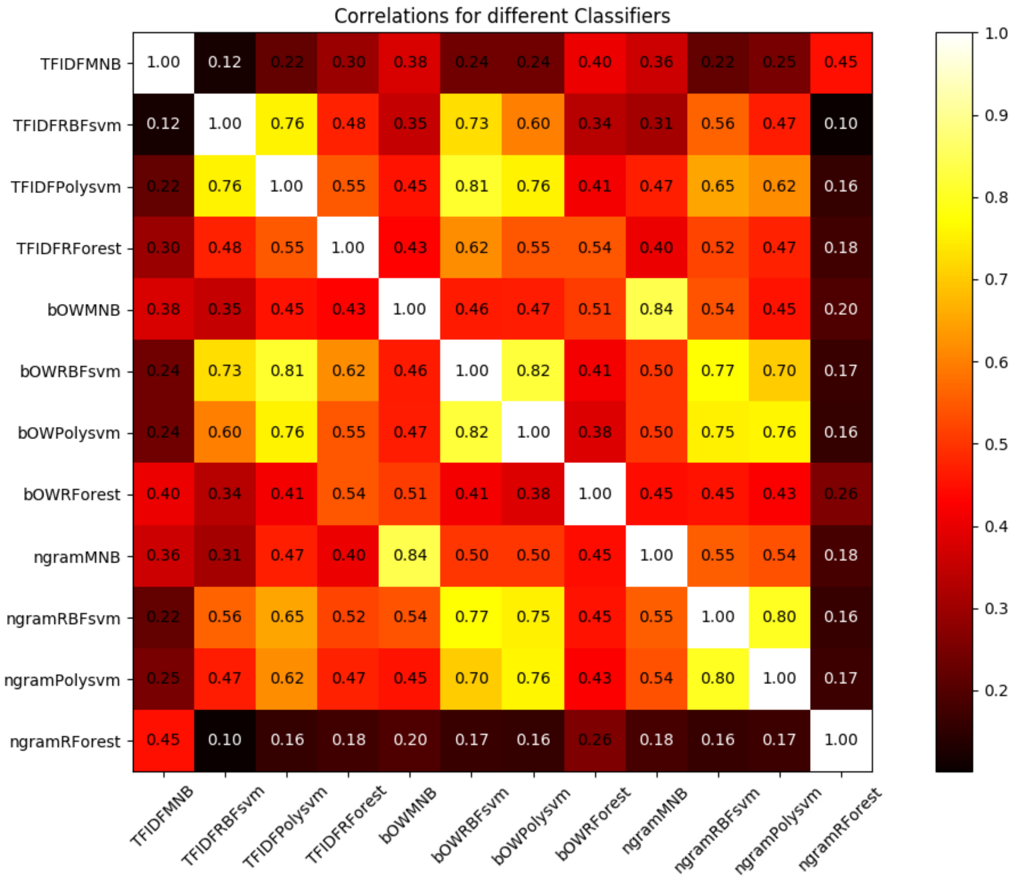
Figure 2: F1 score with mean average rule on minor category

1.3 correlations of different classifiers

A combination of the output of other predictors is only useful if they disagree on some inputs. Figures 3 shows the correlations for different Classifiers, it is a symmetric matrix. Poly SVM and RBF SVM with bag-of-word features are highly correlated with those with TFIDF features, the combination won't get significant benefit. But MMB (multinomial Naive Bayes classifier) with TFIDF is not correlated with the SVMs (with any other features). See the first column of the matrix (figure 3).

The ensemble model gets improvement, and some experiments show that outperforms every single one within the ensemble model. See appendix table 4

Figure 3: Correlations for different Classifiers



	TFIDFMNB	TFIDFRBFsvm	TFIDFPolysvm	TFIDFRForest	bOWMNB	bOWRBFsvm	bOWPolysvm	bOWRForest	ngramMNB	ngramRBFsvm	ngramPolysvm	ngramRForest
TFIDFMNB	1.00	0.12	0.22	0.30	0.38	0.24	0.24	0.40	0.36	0.22	0.25	0.45
TFIDFRBFsvm	0.12	1.00	0.76	0.48	0.35	0.73	0.60	0.34	0.31	0.56	0.47	0.10
TFIDFPolysvm	0.22	0.76	1.00	0.55	0.45	0.81	0.76	0.41	0.47	0.65	0.62	0.16
TFIDFRForest	0.30	0.48	0.55	1.00	0.43	0.62	0.55	0.54	0.40	0.52	0.47	0.18
bOWMNB	0.38	0.35	0.45	0.43	1.00	0.46	0.47	0.51	0.84	0.54	0.45	0.20
bOWRBFsvm	0.24	0.73	0.81	0.62	0.46	1.00	0.82	0.41	0.50	0.77	0.70	0.17
bOWPolysvm	0.24	0.60	0.76	0.55	0.47	0.82	1.00	0.38	0.50	0.75	0.76	0.16
bOWRForest	0.40	0.34	0.41	0.54	0.51	0.41	0.38	1.00	0.45	0.45	0.43	0.26
ngramMNB	0.36	0.31	0.47	0.40	0.84	0.50	0.50	0.45	1.00	0.55	0.54	0.18
ngramRBFsvm	0.22	0.56	0.65	0.52	0.54	0.77	0.75	0.45	0.55	1.00	0.80	0.16
ngramPolysvm	0.25	0.47	0.62	0.47	0.45	0.70	0.76	0.43	0.54	0.80	1.00	0.17
ngramRForest	0.45	0.10	0.16	0.18	0.20	0.17	0.16	0.26	0.18	0.16	0.17	1.00

(MMB short for (multinomial Naive Bayes classifier), bOW short for bag-of-word features, Rorest short for random Forest. Apply for the whole report.)

Classifier	F1	Classifier	F1
BagOfWord RBFsvm	0.52	BagOfWord RBFsvm	0.52
TFIDF **Polysvm**	0.39	TFIDF MMB	0.43
Ensemble model	0.42	Ensembel	0.522
		Ngram RBF svm	0.53
		TFIDF MMB	0.43
		Ensemble model	0.54
Classifiers have **high** correlation		Classifiers have **low** correlation	

Table 1: experiments of Classifiers Correlations affect the ensemble model

1.4 the final model

After using Cross-validation grid-search find out the best parameters of individual model. Then tried with deferent combinations of with classifiers. Using Clf.predict_proba() of individual model to get the probabilities then feed to ensemble model to do averaging with the mean combination rule and weights. Grid-search was used to find the optimal average weights.

It turn out that the highest F1 score is the model with ngram features, ngram_range=(1, 3), mean combination rule, 4 classifiers combination: Random Forest, PolySVM, RBF SVM, multinomial Naive Bayes. Table 5 shows the hyperparameter selection procedure. And chi2 feature selection was uesd to select the best 81 percent of features.

Classifier	Parameter search space	Best Parameter find with Grid search
MultinomialNB	Alpha:{0.0001,...10}	Alpha:4.44
SVM-RBF	C, log space :{0.001...50}, gamma log space:{0.0001,⋯,10}	C:27.5, gamma:0.0015, class_weight:balanced
SVM-Poly	C, log space :{0.001...50}, gamma log space:{0.0001,⋯,10}	C:30, degree:5, gamma:0.001, class_weight: balanced,
RandomForest	min_samples_leaf: {1,2,4,8}, n_estimators:{2,4,8,20,40...,80}, max_features: {1,⋯,123}, min_samples_split:{2,4,8,...50},	bootstrap: False, min_samples_leaf: 8, n_estimators: 20, max_features: 123, criterion:gini, min_samples_split:4, max_depth: None
Ensemble model	Combination weight {0:4}{0:4}{0:4}{0:4}	Weight: 1.02, 1.1, 1.08, 0.8

Table 2: hyperparameter selections

2 Evaluation

In general, the more data the category it has, the higher F1 score or accuracy it can get. Similar experiments have been done in Milestone 1. The distribution of data is very uneven, for those categories only have very few records, when further be splinted to Training set and test set, those records in test set would have very high chance that uncorrelated to that in the training set, and the confidence of the prediction would be very low as well. This is the main reason for reducing the overall accuracy rate.

2.1 The confidence of the predictions varies across categories

Figure 4 and 5 show the confidence on major category and sub category respectively.

Confusion matrix, and classification report see appendix figure 9, table 5.

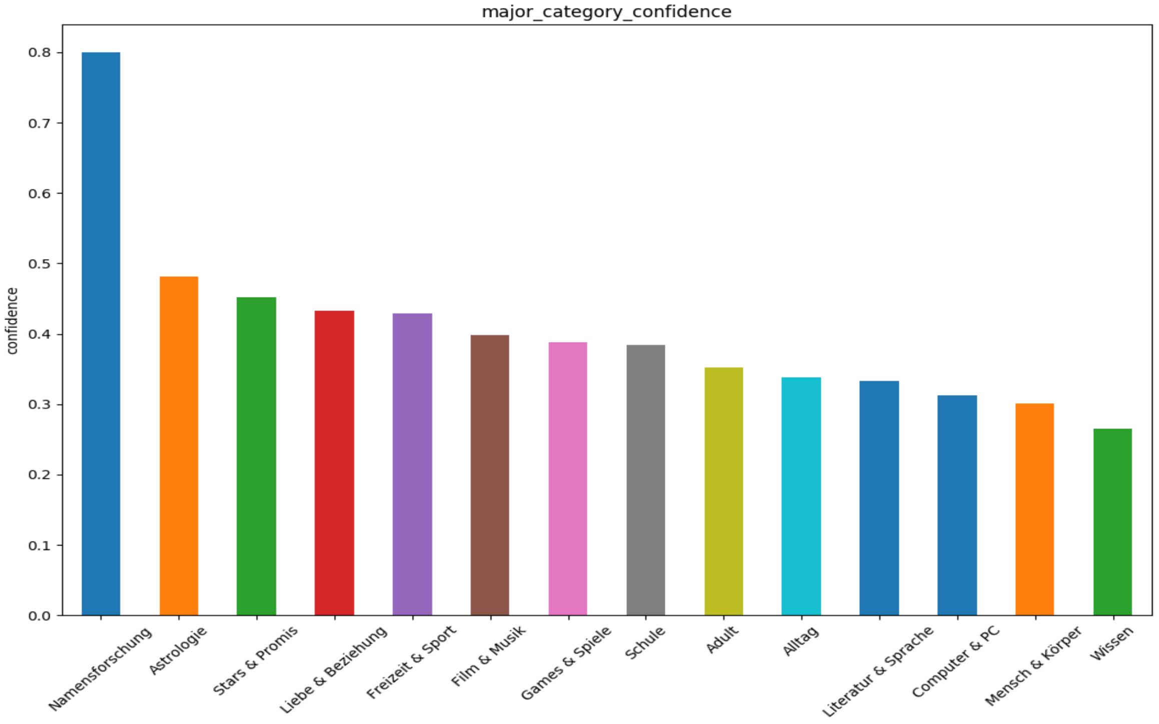


Figure 4: confidence of the predictions on major category (all of 14)

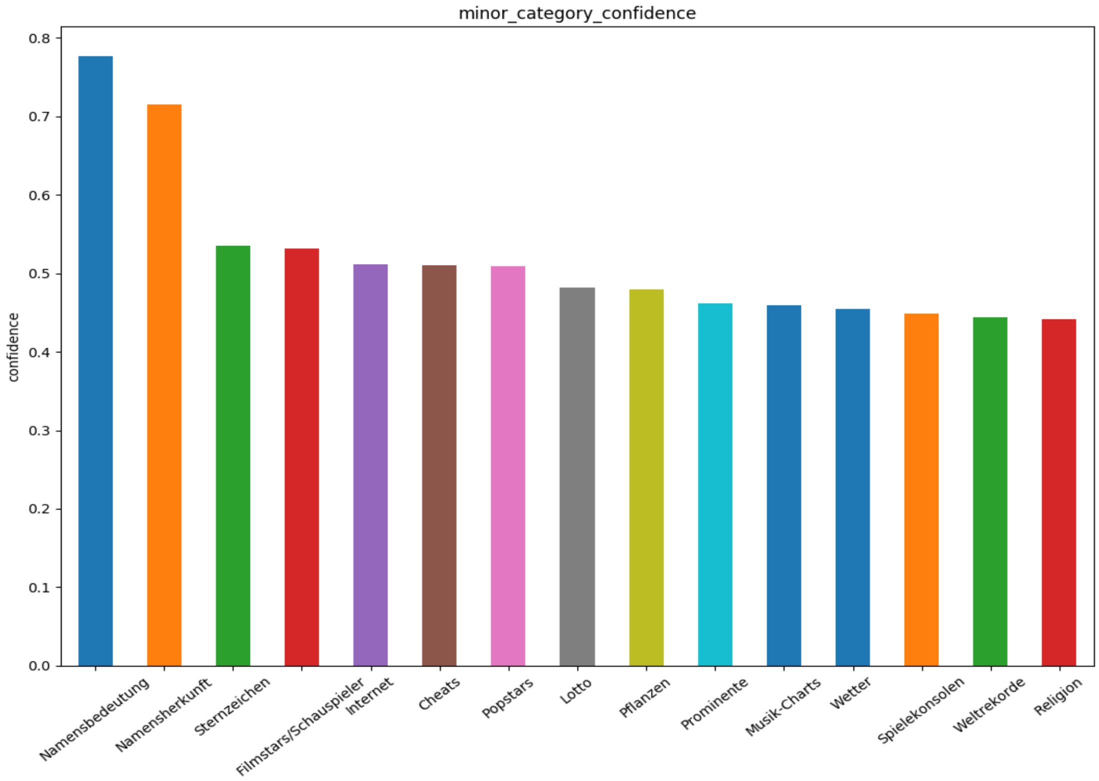


Figure 5: confidence of the predictions on minor category top 15

According to the appendix classification report (table 5) and confusion matrix (appendix figure 9)

In general the higher confidence with relatively higher F1 score, Particularly, category "Namensbedeutung" has the highest confidence score, also with the highest F1 score and can be detected reliably.

2.2 expected time saving

Assume that a human takes **x** seconds = 10 to categorize a question correctly into the main categories. And prediction by the machine learning model takes 0 seconds. However, wrong predictions have to be corrected later, the later correction takes y>x, here assume y =25 seconds.

Table 1 shows the expected time saving when %5 10% 20% wrong categorizations (error rate) are acceptable.

Acceptable error rate	time saving	time saving (with later correction)
5%	2%	2%
10%	8.6%	8.6%
20%	21%	17%

Table 3: expected time saving

Figure 6 shows the trends of expected time saving respect to acceptable error rate. From this figure we can see that there is no significate difference as acceptable error rate is very small, because in this phrase, only those prediction with very high confidence will be accepted, namely seldom questions come to machine learning model, since most of them are categorized by human, no later corrections are needed. As the acceptable error rate increase, the more data predicted with computer, the wrong prediction number grows as well, this require human take y seconds to further stumble upon error.

The trend that with human correction depend on human correction speed **y**, assume that y is very close to x, then the trend is almost no difference with that without further correction, namely, the two lines in figure 6 would be almost the same. However, if human correction time is long, here y is 25 seconds, 2.5 times longer than x, the machine learning model doesn't help to save time, as the acceptable error rate increase, as we can see in the figure, the trend reversed, the percentage close to 0 and become negative (means it's a wasted of time), it is better to let human to do everything without machine learning model.

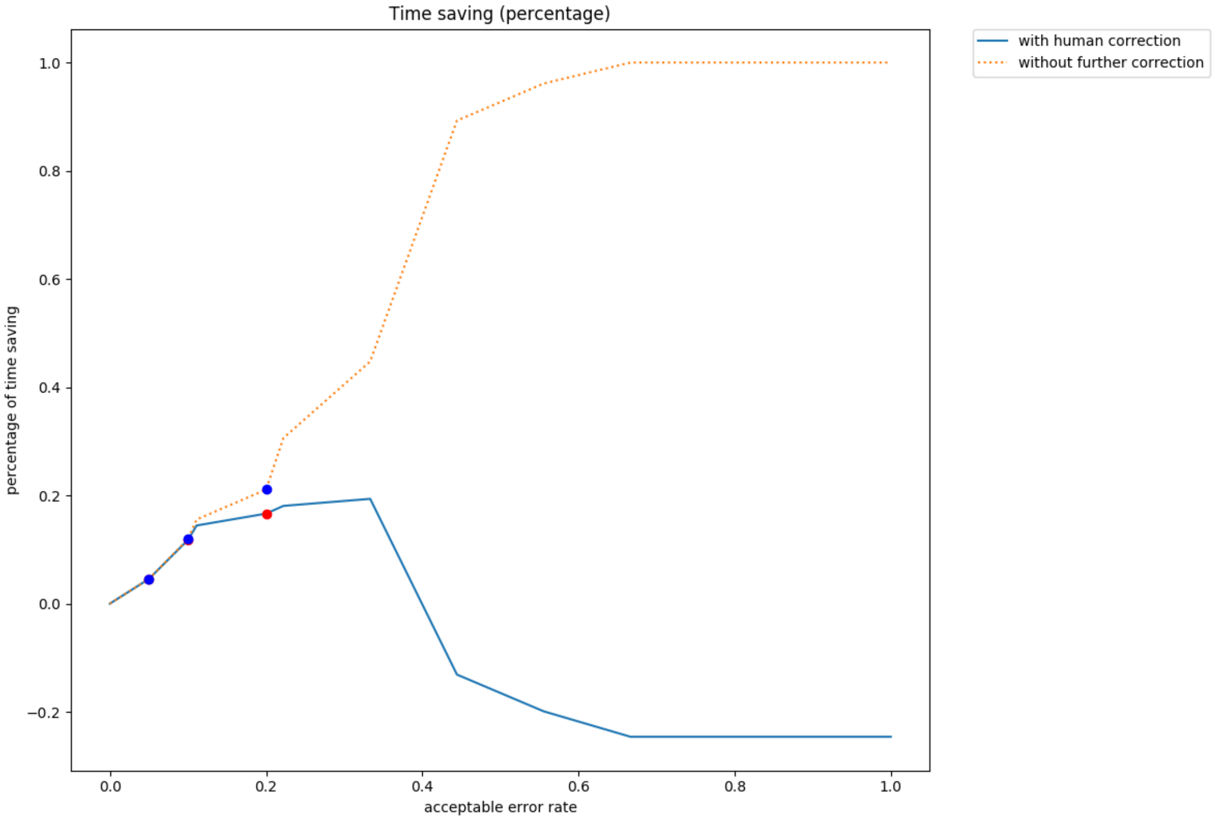
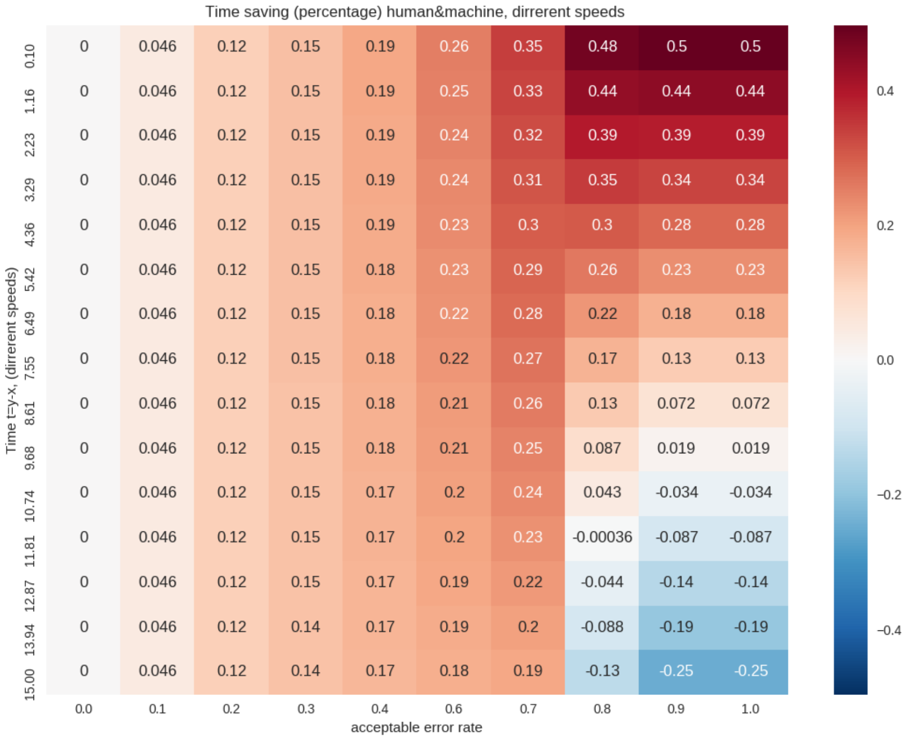


Figure 6: expected time saving

Consider different human correction speed, here, y = times * x, times from 1.01 to 2.5. The possible trends with t=y-x and error rate combinations can be seemed in figure 7. As the acceptable error rate incase to very high, the machine leaning model will produce lots of error predictions that require human to further correction, in those phase, values become negative, computer doesn't save time at all.

In conclusion, the machine learning model is useful in practice, can save remarkable time when the acceptable error rate is relative low (around 20%) and human correction speed y is not very long(y/x<2.5). It might be a good idea to take the individual category into account since some categories can be detected much more reliably, like the categories “Namensbedeutung” and “Games & Spiele” have relative higher F1 score.



	0.0	0.1	0.2	0.3	0.4	0.6	0.7	0.8	0.9	1.0
0.10	0	0.046	0.12	0.15	0.19	0.26	0.35	0.48	0.5	0.5
1.16	0	0.046	0.12	0.15	0.19	0.25	0.33	0.44	0.44	0.44
2.23	0	0.046	0.12	0.15	0.19	0.24	0.32	0.39	0.39	0.39
3.29	0	0.046	0.12	0.15	0.19	0.24	0.31	0.35	0.34	0.34
4.36	0	0.046	0.12	0.15	0.19	0.23	0.3	0.3	0.28	0.28
5.42	0	0.046	0.12	0.15	0.18	0.23	0.29	0.26	0.23	0.23
6.49	0	0.046	0.12	0.15	0.18	0.22	0.28	0.22	0.18	0.18
7.55	0	0.046	0.12	0.15	0.18	0.22	0.27	0.17	0.13	0.13
8.61	0	0.046	0.12	0.15	0.18	0.21	0.26	0.13	0.072	0.072
9.68	0	0.046	0.12	0.15	0.18	0.21	0.25	0.087	0.019	0.019
10.74	0	0.046	0.12	0.15	0.17	0.2	0.24	0.043	-0.034	-0.034
11.81	0	0.046	0.12	0.15	0.17	0.2	0.23	-0.00036	-0.087	-0.087
12.87	0	0.046	0.12	0.15	0.17	0.19	0.22	-0.044	-0.14	-0.14
13.94	0	0.046	0.12	0.14	0.17	0.19	0.2	-0.088	-0.19	-0.19
15.00	0	0.046	0.12	0.14	0.17	0.18	0.19	-0.13	-0.25	-0.25

Figure 7: expected time saving that consider different speeds

2.3 test error and rejection rates.

Figure 8 shows the trend that the higher confidence the classifier it has, the lower error score it will get. Threshold C control that only classifier confidence larger than C will accepted, in other words, threshold C=0, all predictions are done by classifier. As threshold C increased, in general the two green lines (error rate on major and sub categories) decreased, in the last phase, threshold close to 1, still has numbers of predictions close 1, but made wrong predictions although with 1 confidence score. But the blue and red lines (rejection rates on major and sub categories) increased, means, lots of prediction confidences lower than the C, and ignored the prediction.

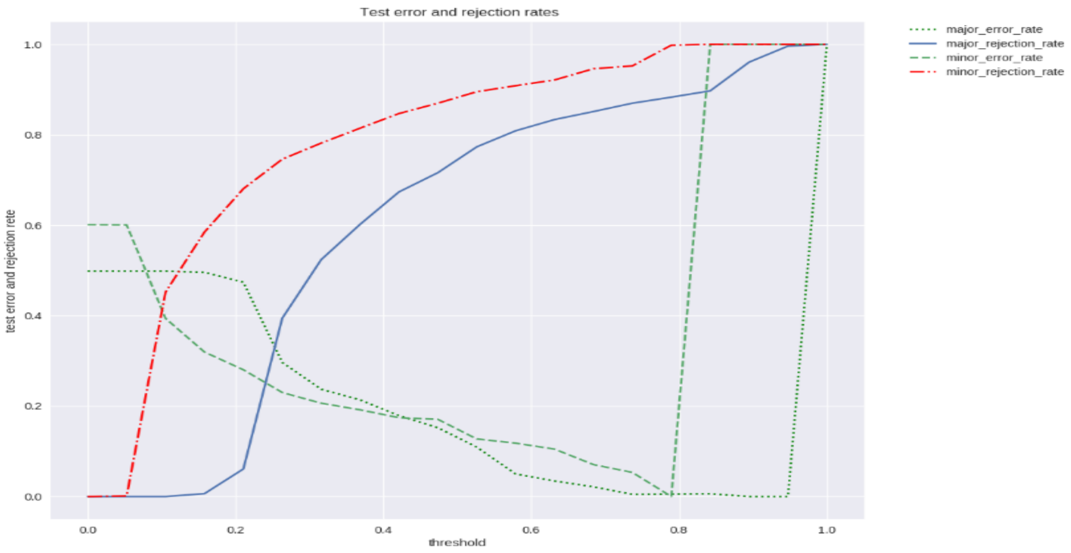


Figure 8: test error and rejection rates.

2.4 Further improvement

a) Try to get as more training data as possible, there is study [3] shows more data usually beats better algorithms. As mentioned before, in general, some categories have very few training data and got high test error, as the training data size increase, the accuracy increase.

b) Defining the category distance that measure the distance or similarity between two categories base on the input feature x. A classifier will get more punishment score if it classifies a question to more unrelated class. Since there are some classes are highly overlapped or very similar. In practice, classifying to a similar category sometime is acceptable, and has higher availability compared to classifier to totally unrelated category. Using this scoring method would improve the classifier quality.

c) Analysis of the words that contribute to the overall classification decision. (see e.g. [4]).

In addition to predict the question's category very accurately, it is also highly desirable to understand how and why the categorization process takes place.

We can score the individual word in question that indicate how largely it contribute to the overall classification decision. Identifying relevant words in classification decision could contribute to the design of more accurate and efficient classifiers. If a classifier make decision based on totally unrelated words (from human knowledge and intuition's perspective）, although the model achieved relative higher accuracy in test set, but we would expect that it will generalize very bad on future dataset if the destitution changes, especially the available training data is limited. This method would help us to inspect the potential issue of the classifier.

2.5 Further ideas on ML projects in the SMS Guru context

a) feature augmentation

Using possible feature augmentation to enhance the machine learning model with generated features based on domain-specific and common-sense knowledge. For example, make use of Open Directory, word Net, Semantic Web to represent knowledge and concepts.

b) Name entity Recognition

Named Entity Recognition sifts through text data and locates noun phrases called named entities. Named entities can then be organized under predefined categories, such as "person," "organization," "location," "number," or "duration." NER also can helps a lot in disambiguation, that is, identifying the right entity among a number of entities with the same names. For example, "apple" standing for both "Apple, Inc." the company and the fruit. There are some studies [2] show this can improve the classification performance.

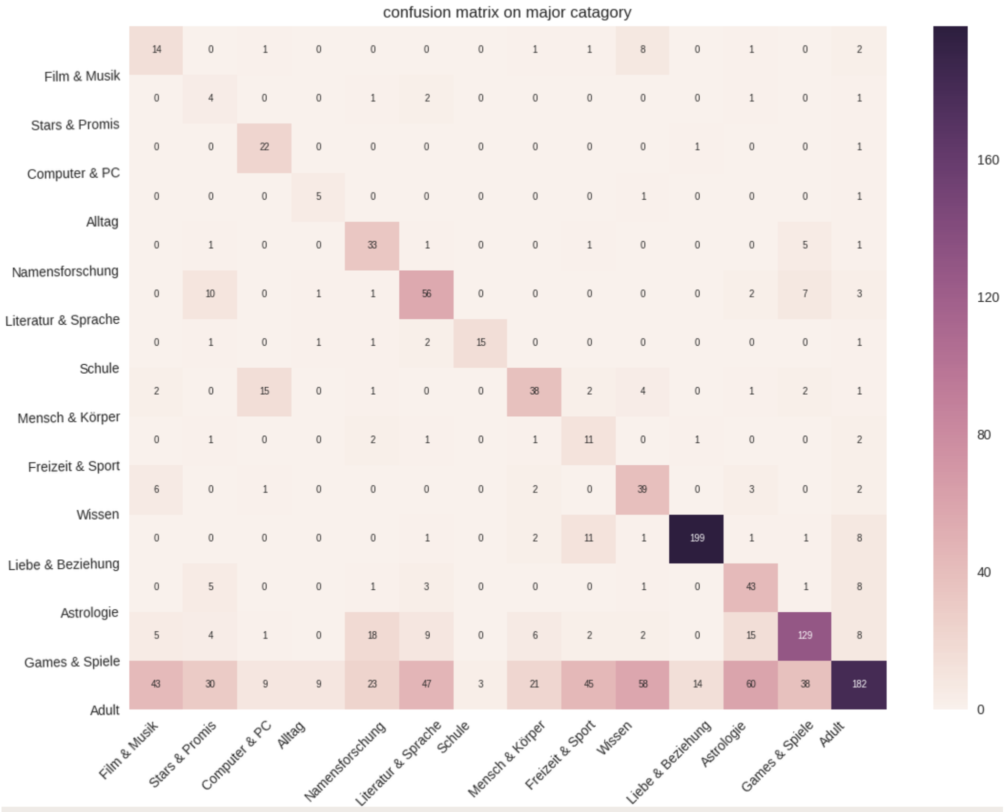
C) Using Neural Network to find optimal ensemble way.

Finding the optimal weights [5] for the individuals of the ensemble is essential. Mean and production combination rule are less flexible. Make use of neural network to fusion the different features (TFIDF, bag of word, word embedding …), do ensemble base on the category level, namely different category has deferent weights, because some model with some features are only good at specific category but performs very bad on other categories.

Appendix

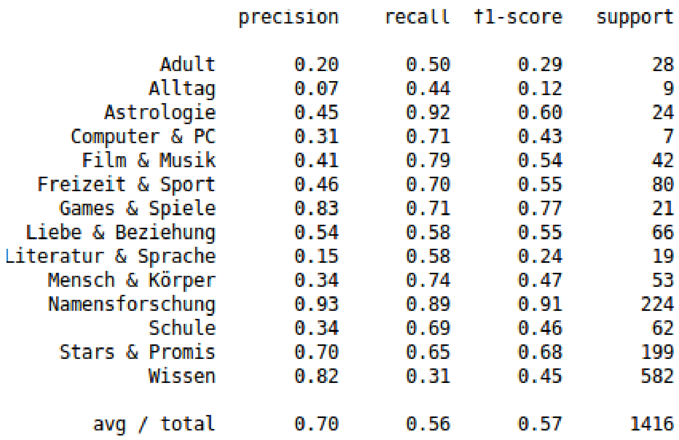
ngram	f1 score	Average improvement
major MNB	0.54	
major RBFsvm	0.53	
major Polysvm	0.56	
major Random Forest	0.25	
major ensemble (mean)	0.57	+0.0975 ±0.002
major ensemble(product)	0.36	
minor MNB	0.37	
minor RBFsvm	0.35	
minor Polysvm	0.41	
minor Random Forest	0.12	
minor ensemble (mean)	0.44	+0.13 ±0.003
minor ensemble (product)	0.26	
bagOfWord		
major MNB	0.53	
major RBFsvm	0.52	
major Polysvm	0.57	
major Random Forest	0.49	
major ensemble (mean)	0.56	
major ensemble (product)	0.52	
minor MNB	0.35	
minor RBFsvm	0.33	
minor Polysvm	0.4	
minor Random Forest	0.37	
minor ensemble (mean)	0.42	
minor ensemble (product)	0.38	
TFIDF		
major MNB	0.43	
major RBFsvm	0.39	
major Polysvm	0.53	
major Random Forest	0.5	
major ensemble(mean)	0.54	
minor MNB	0.22	
minor RBFsvm	0.2	
minor Polysvm	0.32	
minor Random Forest	0.37	
minor ensemble(mean)	0.38	

Table 4: F1 score of different models



	Film & Musik	Stars & Promis	Computer & PC	Alltag	Namensforschung	Literatur & Sprache	Schule	Mensch & Körper	Freizeit & Sport	Wissen	Liebe & Beziehung	Astrologie	Games & Spiele	Adult
Film & Musik	14	0	1	0	0	0	0	1	1	8	0	1	0	2
Stars & Promis	0	4	0	0	1	2	0	0	0	0	0	1	0	1
Computer & PC	0	0	22	0	0	0	0	0	0	0	1	0	0	1
Alltag	0	0	0	5	0	0	0	0	0	1	0	0	0	1
Namensforschung	0	1	0	0	33	1	0	0	1	0	0	0	5	1
Literatur & Sprache	0	10	0	1	1	56	0	0	0	0	0	2	7	3
Schule	0	1	0	1	1	2	15	0	0	0	0	0	0	1
Mensch & Körper	2	0	15	0	1	0	0	38	2	4	0	1	2	1
Freizeit & Sport	0	1	0	0	2	1	0	1	11	0	1	0	0	2
Wissen	6	0	1	0	0	0	0	2	0	39	0	3	0	2
Liebe & Beziehung	0	0	0	0	0	1	0	2	11	1	199	1	1	8
Astrologie	0	5	0	0	1	3	0	0	0	1	0	43	1	8
Games & Spiele	5	4	1	0	18	9	0	6	2	2	0	15	129	8
Adult	43	30	9	9	23	47	3	21	45	58	14	60	38	182

Figure 9: confusion matrix on major category



```
                     precision    recall  f1-score   support

              Adult       0.20      0.50      0.29        28
             Alltag       0.07      0.44      0.12         9
         Astrologie       0.45      0.92      0.60        24
      Computer & PC       0.31      0.71      0.43         7
       Film & Musik       0.41      0.79      0.54        42
   Freizeit & Sport       0.46      0.70      0.55        80
     Games & Spiele       0.83      0.71      0.77        21
  Liebe & Beziehung       0.54      0.58      0.55        66
Literatur & Sprache       0.15      0.58      0.24        19
    Mensch & Körper       0.34      0.74      0.47        53
    Namensforschung       0.93      0.89      0.91       224
             Schule       0.34      0.69      0.46        62
     Stars & Promis       0.70      0.65      0.68       199
             Wissen       0.82      0.31      0.45       582

        avg / total       0.70      0.56      0.57      1416
```

Table 5: Classification report on major category

References

[1] Combining multiple classifiers by averaging or by multiplying? In: Pattern recognition 33 (2000), Nr. 9, S. 1475–1485

[2] Using Named Entity Recognition as a Classification Heuristic

[3] http://anand.typepad.com/datawocky/2008/03/more-data-usual.html

[4] "What is relevant in a text document?": An interpretable machine learning approach

[5] Neural Network Ensembles, Cross Validation, and Active Learning