UKARA 1.0 Challenge Track 1: Automatic Short-Answer Scoring in Bahasa Indonesia

Ali Akbar Septiandri
Airy
Jakarta, Indonesia
ali.septiandri@airy.com

Yosef Ardhito Winatmoko
Jheronimus Academy of Data Science
’s-Hertogenbosch, The Netherlands
y.a.winatmoko@uvt.nl

Abstract

We describe our third-place solution to the UKARA 1.0 challenge on automated essay scoring. The task consists of a binary classification problem on two datasets — answers from two different questions. We ended up using two different models for the two datasets. For task A, we applied a random forest algorithm on features extracted using unigram with latent semantic analysis (LSA). On the other hand, for task B, we only used logistic regression on TF-IDF features. Our model results in F1 score of 0.812.

1 Introduction

Automated essay scoring is the application of computers technologies to assist human grader in evaluating the score of written answers (Dikli, 2006). The first track of UKARA 1.0 is the binary classification version of essay scoring, where participants are expected to develop a model that can distinguish right and wrong answers in free text format. The organizer published the questions, the responses with the labels, and the guideline on how to determine whether an answer is acceptable.

During a period of five weeks, the training set and the development set were available and we could validate our model through the score of the development set in a leaderboard. Subsequently, the test set was released, which consists of roughly four times the size of the development set. We are required to submit predicted labels based on the model that we have developed and the winner was determined by the F1 score of the submitted prediction.

Our final submission to this task consists of feature extraction, such as n-grams and TF-IDF, and classical machine learning algorithms, namely logistic regression and random forest. We did not use deep learning at all in our submission. The details of the dataset and our approach will be discussed in the following sections.

2 Datasets

The dataset consists of two questions and the respective responses collected by the organizer of the challenge. All questions and responses are in Indonesian. The first question, from now on will be referred as task A, asked about the consequence of the climate change. Concretely, what are the potential problems faced by a climate refugee when they have to migrate to a new place. The second question, referred as task B, is based on an experiment. There were potential customers, initially wanted to buy clothes, who prefer to donate the money instead when they are presented with videos of the clothes manufacturing worker condition before paying. The respondents were required to give their opinion on why do people decided to change their mind. The statistics of the responses for both tasks are shown in Table 1.

	Task A	Task B
#Positive Train	191(71%)	168(55%)
#Negative Train	77(29%)	137(45%)
Avg. #Char	87.23	97.33
#Dev	215	244
#Test	855	974

Table 1: Summary statistics of the dataset.

3 Methodology

3.1 Preprocessing

For the preprocessing steps, we first tokenized and lemmatized the text using bahasa Indonesia tokenizer provided by spaCy (Honnibal and Montani, 2017). We then extracted the features using bag-of-words or TF-IDF. Since the resulting matrix from this feature extraction method tends to be sparse, and to encode token relations, we applied Latent Semantic Analysis (LSA) using Singular Value Decomposition (SVD) (Deerwester et al., 1990) on the matrix.

Based on our observation, we noticed that the labels of the provided training set are highly inconsistent. Some responses are clearly labelled incorrectly. For illustration, in task A we found "*untuk pindah ke daerah yang aman*" (to move to a safe place) labelled as 1 (correct) while clearly it does not fit the criteria based on the guideline. The mislabeling was even more prominent in task B: "*karana dengan menyumbang kita bisa membuat produksi pakaian menjadi lebih beretika*" (By donating, we can make clothes production becomes more ethical) is considered wrong while "*agar upaya untuk membuat produksi pakaian menjadi lebih beretika.*" (As an effort to make clothes production becomes more ethical) is approved. To approach this problem, we decided to prepare a separate training set with manually corrected labels based on our own judgment. The correction result is shown in Table 2.

Finally, as the responses contain a lot of typo and slang words, we also experimented with simple typo corrector using python difflib package and Indonesian colloquial dictionary (Salsabila et al., 2018). We tried every possible combination of preprocessing steps and whether to use altered version of the training set with a parameter optimization library described in the following subsection.

	Original Label	**Corrected Label**	**Count**
Task A	0	1	10
	1	0	4
Task B	0	1	46
	1	0	13

Table 2: Corrected labels of the training set.

3.2 The Winning Approach

After trying several machine learning algorithms, such as k-Nearest Neighbors, Naïve Bayes, logistic regression, and random forest, we found that random forest was the best model for task A. This corroborates what was found by Fernández-Delgado et al. (2014) in their comprehensive comparisons among several machine learning algorithms on different datasets. On the other hand, logistic regression with L2 regularization was the best for task B. The machine learning library used in this study is scikit-learn (Pedregosa et al., 2011). Since the dataset is quite small, we used 5-fold cross validation on the training set to avoid overfitting. For our winning approach, we set the `n_estimators` parameter for random forest to 200 and keep the default values for other parameters. We also found that it is the best to keep the default parameter values for the logistic regression model.

3.3 Alternative Approaches

In parallel, we also experimented with optimization using hyperopt[1] library, which utilizes sequential model-based optimization (Bergstra et al., 2011). We tried to optimize the hyperparameters, including the preprocessing steps, of four different machine learning algorithms: logistic regression, random forest, gradient boosting tree, and support vector machine. We trained separate models for task A and task B. In addition, we tested a voting-based ensemble model by combining all the optimized model of each algorithm. The evaluation metric used for the optimization, including for the voting-ensemble model, is F1 score. The results of the experiments are presented in the next section along with the discussion.

4 Results and Discussion

We found that choosing different preprocessing methods resulted in different performances in the two tasks. Therefore, we varied the use of unigram or TF-IDF, and whether we should apply SVD to the resulting matrix. On the other hand, we found that it is always better to use the lemmatizer built on top of spaCy in this task. Moreover, removing stopwords did not contribute much to the performance on the training set. Table of local CV results can be seen in Table 3 and Table 4.

For this challenge, we need to optimize the F1 score. Therefore, it is clear from Table 4 that we should use the model from TF-IDF with random forest algorithm for task B. From what we can see ini Table 3, we should also go with the same method for task A. However, we decided to be more pessimistic by looking at the largest F1 score after subtracting 1 standard deviation from the mean F1. Thus, we chose 1-gram + SVD + random forest for task A.

[1] http://hyperopt.github.io/hyperopt/

	Precision	**Recall**	**F1**
1-gram+RF	0.845 ± 0.057	0.921 ± 0.037	0.881 ± 0.035
1-gram+logreg	0.857 ± 0.074	0.869 ± 0.067	0.862 ± 0.065
1-gram+SVD+RF	0.794 ± 0.025	0.984 ± 0.014	0.879 ± 0.014
1-gram+SVD+logreg	$\mathbf{0.859 \pm 0.069}$	0.874 ± 0.047	0.866 ± 0.053
TF-IDF+RF	0.847 ± 0.054	0.942 ± 0.039	$\mathbf{0.891 \pm 0.036}$
TF-IDF+logreg	0.748 ± 0.019	0.979 ± 0.034	0.848 ± 0.022
TF-IDF+SVD+RF	0.772 ± 0.025	$\mathbf{0.990 \pm 0.014}$	0.867 ± 0.014
TF-IDF+SVD+logreg	0.751 ± 0.025	0.979 ± 0.034	0.850 ± 0.025

Table 3: 5-fold cross validation results from task A

	Precision	**Recall**	**F1**
1-gram+RF	0.731 ± 0.066	0.744 ± 0.054	0.736 ± 0.047
1-gram+logreg	$\mathbf{0.735 \pm 0.046}$	0.727 ± 0.080	0.730 ± 0.059
1-gram+SVD+RF	0.686 ± 0.035	0.762 ± 0.046	0.721 ± 0.032
1-gram+SVD+logreg	0.722 ± 0.067	0.726 ± 0.089	0.723 ± 0.074
TF-IDF+RF	0.709 ± 0.036	0.750 ± 0.049	0.728 ± 0.034
TF-IDF+logreg	0.725 ± 0.035	0.810 ± 0.060	$\mathbf{0.764 \pm 0.035}$
TF-IDF+SVD+RF	0.637 ± 0.026	$\mathbf{0.834 \pm 0.061}$	0.721 ± 0.036
TF-IDF+SVD+logreg	0.705 ± 0.023	0.809 ± 0.063	0.753 ± 0.036

Table 4: 5-fold cross validation results from task B

Table 5 shows the performance of our best single models compared with two alternative approaches. First, we used the optimized ensemble model trained on the original training set (Ens+Ori). Second, we also use a similar method but with the label-corrected training set (Ens+Upd). While we can see in Table 5 that the F1 score on task B is higher on the label-corrected training set and got a similar result for the development set, the test score is lower than the best single models. The test set labels were most likely as noisy as the training set, thus making the training score with modified label not representative.

To analyze how hard it is to separate the right from the wrong answers, we reduced the dimensionality of the data into 2D using 1-gram, SVD, and t-SNE (Maaten and Hinton, 2008). Figure 1 suggests that it is harder to separate the two answers in task B. Since most of the answers are short, we also cannot see "islands" from applying t-SNE to the data.

A similar problem was also shown in Figure 2 and Figure 3. We can see a lot more data points in the 0.4-0.6 prediction range in Figure 3. This suggests more uncertainty in the model. We argue that this is potentially because of the incorrect labels in the original training set for task B.

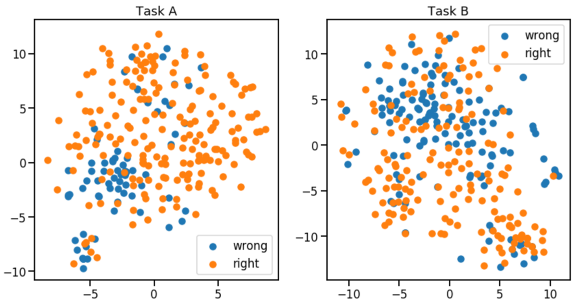

Figure 1: t-SNE visualisation

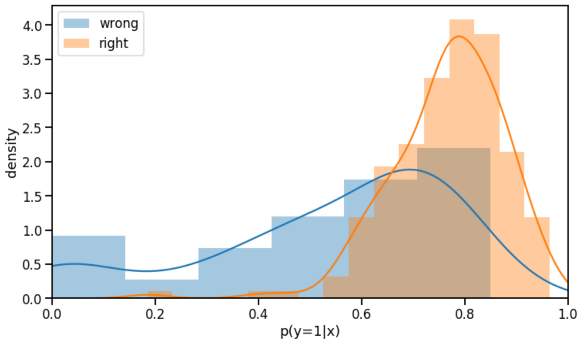

Figure 2: Best model prediction (random forest) with probability on Task A

	Train A	Train B	Dev	Test
Best	0.879	0.764	0.810	0.812
Ens+Ori	0.885	0.764	0.799	0.801
Ens+Upd	0.898	0.831	0.810	0.803

Table 5: F1 score comparison with the alternative Ensemble model.

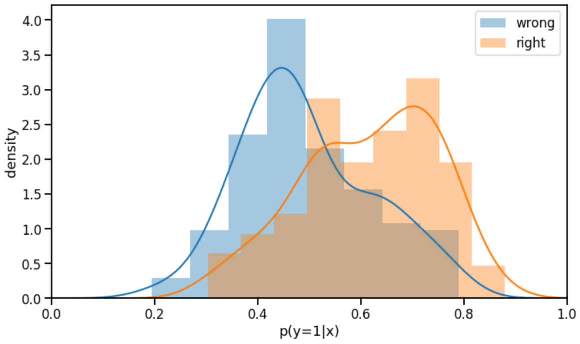

Figure 3: Best model prediction (logistic regression) with probability on Task B

5 Conclusions

In this report, we describe our winning approach for UKARA 1.0 Challenge Track 1. During the competition, we experimented with single models and ensemble models to predict which answer is correct given an open-ended question. We also tried to re-label the training set and pre-process with text correction which gave a boost in our local cross validation. However, we found that single models with less pre-processing performed better in the test set. The best single model for task A is random forest with unigram+SVD and for task B is logistic regression with TF-IDF. The prediction of both model achieved an overall F1 score of 0.812, which was enough to get us the third position in the final leaderboard.

References

James S Bergstra, Rémi Bardenet, Yoshua Bengio, and Balázs Kégl. 2011. Algorithms for hyper-parameter optimization. In *Advances in neural information processing systems*, pages 2546–2554.

Scott Deerwester, Susan T Dumais, George W Furnas, Thomas K Landauer, and Richard Harshman. 1990. Indexing by latent semantic analysis. *Journal of the American society for information science*, 41(6):391–407.

Semire Dikli. 2006. An overview of automated scoring of essays. *The Journal of Technology, Learning and Assessment*, 5(1).

Manuel Fernández-Delgado, Eva Cernadas, Senén Barro, and Dinani Amorim. 2014. Do we need hundreds of classifiers to solve real world classification problems? *The Journal of Machine Learning Research*, 15(1):3133–3181.

Matthew Honnibal and Ines Montani. 2017. spaCy 2: Natural language understanding with Bloom embeddings, convolutional neural networks and incremental parsing. To appear.

Laurens van der Maaten and Geoffrey Hinton. 2008. Visualizing data using t-sne. *Journal of machine learning research*, 9(Nov):2579–2605.

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. 2011. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

Nikmatun Aliyah Salsabila, Yosef Ardhito Winatmoko, Ali Akbar Septiandri, and Ade Jamal. 2018. Colloquial indonesian lexicon. In *2018 International Conference on Asian Language Processing (IALP)*, pages 226–229. IEEE.

A Supplemental Material

The code for this analysis can be seen on `https://github.com/aliakbars/ukara/`.