Ensemble Methods to Distinguish Mainland and Taiwan Chinese

Hai Hu*† Wen Li*† He Zhou*† Zuoyu Tian† Yiwen Zhang† Liang Zou‡
Department of Linguistics, Indiana University†
Courant Institute of Mathematical Sciences, New York University‡
{huhai,wl9,hzh1,zuoytian,yiwezhan}@iu.edu, lz1904@nyu.edu

Abstract

This paper describes the IUCL system at VarDial 2019 evaluation campaign for the task of discriminating between Mainland and Taiwan variation of mandarin Chinese. We first build several base classifiers, including a Naive Bayes classifier with word *n*-gram as features, SVMs with both character and syntactic features, and neural networks with pre-trained character/word embeddings. Then we adopt ensemble methods to combine output from base classifiers to make final predictions. Our ensemble models achieve the highest F1 score (0.893) in simplified Chinese track and the second highest (0.901) in traditional Chinese track. Our results demonstrate the effectiveness and robustness of the ensemble method.

1 Introduction

Like many other languages in the world, Mandarin has several varieties among different speaking communities, mainland China, Taiwan, Malaysia, Indonesia, etc. Previous research on these varieties are mainly focused on language differences and integration (Yan-bin, 2012). Discriminating between the Mainland and Taiwan variations of Mandarin Chinese (DMT) is one of the shared tasks at VarDial evaluation campaign 2019, aiming to determine if a given sentence is taken from news articles from Mainland China or Taiwan (Zampieri et al., 2019). This task not only serves as a platform to test various models, but also encourages linguists to rethink the different linguistic features among those varieties.

This paper describes the IUCL (Indiana University Computational Linguistics) systems and submissions at VarDial 2019. We first build several base classifiers: a Naive Bayes classifier with word *n*-gram as features, Support Vector Machines (SVM) using both character and syntactic features, and neural networks such as LSTM and BERT. We then build ensemble models by using the maximum probability among all base classifiers, or choosing the class with maximum average probability, or training another SVM on top of the output of base classifiers. We apply the three ensemble models for both the simplified Chinese track and the traditional Chinese track, which also correspond to our three submissions on both tracks. As shown in the official evaluation results, our SVM ensemble is ranked the first place on the simplified Chinese test data with a macro-averaged F1 score of 0.893, and our ensemble model using maximum probability from base classifiers ranked second on the traditional Chinese test data with a macro-averaged F1 score of 0.901.

In this paper, we will briefly review related work in Section 2, describe our single classifiers and three ensemble methods in Section 3, and finally present and discuss our results in Section 4, with a conclusion in Section 5.

2 Related Work

Discriminating between similar languages (DSL) is one of the main challenges faced by language identification systems (Zampieri et al., 2017, 2015). Since 2014, the DSL shared task provided a platform for researchers to evaluate language identification methods with standard dataset and evaluation methodology. Previous shared tasks on DSL have differentiated a wide range of languages, including similar languages, such as Bosnian, Croatian and Serbian, and one language spoken in different language societies, such as Brazilian vs. European Portuguese. In these shared tasks, SVMs are probably the most widely used classifier, while logistic regression and naive Bayes also performed well. More recently, Convolutional Neural Networks and Recurrent Neural Networks are also

*Equal contribution

Proceedings of VarDial, pages 165–171
Minneapolis, MN, June 7, 2019

implemented with byte-level, character-level or word-level embeddings. Regarding features in this task, character and word *n*-grams are most frequently used (Zampieri et al., 2015; Malmasi et al., 2016; Zampieri et al., 2017, 2018). Besides, ensemble methods are also used widely in DSL to improve results beyond those of the individual base classifiers, for example, majority voting between a classifier trained with different features, majority voting to combine several classifiers, polarity voting with SVMs, etc. (Jauhiainen et al., 2018).

Like many languages studied in the literature of DSL, Mandarin Chinese also has several varieties among its speaking communities. Previous work on this was done by proposing a top-bag-of-word similarity measures for classifying texts from different variants of the same language (Huang and Lee, 2008). A top-bag-of-word, similarity-based contrastive approach was adopted to solve the text source classification problem. That is, the classification process adopted similar heuristics to generate determined intervals between classes. Then a contrastive elimination algorithm is used that simple majority voting mechanism is employed for determining the final classification results. LDC's Chinese Gigaword Corpus (Huang, 2009) was used as the comparable corpora, which is composed of three varieties of Chinese from Mainland China, Singapore, and Taiwan. Experimentation shows that the contrastive approach using similarity to determine the classes is a reliable and robust tool.

3 Methodology and Data

3.1 Base Classifiers

In this section, we describe each of the classifiers that are included in our final ensemble model.

3.1.1 Naive Bayes Classifier (NB)

Naive Bayes is a simple yet effective probabilistic model that works quite well on various text classification tasks, including sentiment analysis (Narayanan et al., 2013), spam detection (Kim et al., 2006), email classification, and authorship attribution. It has two simplifying assumptions: bag-of-words assumption and conditional independence assumption (Jurafsky and Martin, 2014).

This task can be considered as a binary text classification problem, since an instance is labelled with either M or T. A sentence is treated as a sequence of words, and it is assumed that each word is generated independent of each other. Here we construct features with word unigram, bigram, and trigram, and train a Bernoulli Naive Bayes classifier. Since we have finite number of words, i.e., 18,769 sentences in training set, the classifier is likely to encounter unseen words in development set, therefore smoothing is needed. We iterate the additive smoothing parameter α in range between 0 and 1 for 100 times on training data of both simplified Chinese and traditional Chinese, finding the best α value 0.42 for simplified Chinese and 0.51 for traditional Chinese.

3.1.2 Classifier with Syntactic Features (SYN)

Syntactic features have been shown to be helpful in various text classification tasks, e.g. authorship attribution (Baayen et al., 1996; Gamon, 2004), native language identification (Bykh and Meurers, 2014), and detecting translationese (Hu et al., 2018).

For this task, we employ two types of syntactic features. The first is context-free grammar rules that are not terminal rules. The second is dependency triples (`John_ate_nsubj`). We use the linear SVM classifier from scikit-learn (Pedregosa et al., 2011). The syntactic features are obtained using Stanford CoreNLP with their default models (Manning et al., 2014).

3.1.3 Sequential Model Classifier (SEQ)

We create a classifier based on the multi-layer neutral network model using Keras (Chollet et al., 2015).

First, we prepare the data using bag-of-words model to generate vectors from texts. We create a vocabulary of all the unique words in the test sentences and create a feature vector representing the count of each word. Then we preprocess the data by padding sentences into the same input length of 50.

We build a sequential model including a linear stack of layers. The first layer takes an integer matrix of the size of the vocabulary and shapes the output embedding as 16. The second layer is a global average pooling layer which returns a fixed-length output vector for each example by averaging over the sequence dimension. Then the output vector is piped through a dense layer with 16 hidden units. This model uses the Rectifier activation function, which has been proven to generate good results. The last layer is a dense layer with a single output node using sigmoid activation function. Each value is transformed to a float between 0 and 1, representing probability.

To update weights and find the best parameters in the model, we configure the learning process using the Adam optimizer and the binary cross entropy loss function, which are commonly used for binary classification tasks. Then we train the model by running the iteration for 40 epochs with a batch size of 512.

We use the development set to evaluate the model and make predictions on the test data.

3.1.4 Word-Level Long Short-Term Memory (LSTM)

Long short-term memory is an efficient, gradient-based model (Hochreiter and Schmidhuber, 1997), which is widely used in NLP tasks. We choose as features the most frequent 5000 words in combined training and development data. We transform sentences with one-hot encoder, followed by a 128-dimension embedding layer as well as 64 hidden layers. The batch size and input length are both 32, and we train the model with 7 epochs. To avoid overfitting, we set the dropout rate equal to 0.5.

3.1.5 Classifier with Pretrained Chinese Word Embeddings (EMB)

Word embeddings are dense vector representations of words (Collobert and Weston, 2008; Mikolov et al., 2013). Compared with bag-of-word features, word embeddings are better at capturing semantic and syntactic information in the context. In this task, we choose 300-dimension skip-gram with negative sampling word embeddings trained on People's Daily Newspaper (Li et al., 2018).

The pre-trained embeddings are chosen for the following reasons. First, the training data may not be large enough for building robust word embedding, since subtle semantic relations and infrequent phrases could be overlooked due to the limited data size. The pre-trained embeddings are trained on news over 70 years with 668 million tokens in total, covering the majority of words used in the news genre. Second, the language of People's Daily is quite similar to the Mainland Mandarin data in this task, which also comes from a Chinese official news agency.

We initialise the weights of the projection layer with this 300-dimension pre-trained word embeddings, and keep this layer frozen during the following fine-tuning. Then we feed the output sequentially into a convolutional layer, a max pooling layer and LSTM layer. Finally, two dense layers with sigmoid activation are used.

3.1.6 Bidirectional Encoder Representations from Transformers (BERT)

Bidirectional Encoder Representations from Transformers (Devlin et al., 2018) have shown state-of-art performance in many NLP tasks. The pre-trained model supports multiple languages, and we adapt the Chinese model for classification in this task. Since the model tends to overfit on small dataset, during fine tuning we experiment with 1-3 training epochs. We set the maximum sequence length to be 32, since most of the sentences in training and development data have no more than 32 words. We train the model with two epochs for official submission, since it performs the best when evaluating on development data.

3.2 Ensemble Classifiers

Ensemble models have been widely adopted in various text classification tasks and machine learning applications (e.g., Liu et al., 2016). Ensemble learning can combine weak learners into a strong learner to improve the performance, and it also helps to reduce the variance of models prone to overfitting.

Each classifier outputs the probabilities for the input sentence to be of class M and T. These probabilities are then passed on to an ensemble model which makes the final decision. We experiment with three ensemble models.

MaxProb The MaxProb ensemble simply looks at the max probability of an input sentence. That is, the classifier that is most confident of its decision will have the final say.

MaxMeanProb The MaxMeanProb ensemble computes the average probability for the two classes (M, T) across all classifiers and returns the label with higher average probability.

SVM Classifier stacking is often used in ensemble learning to integrate different models to produce the final output (e.g., Li and Zou, 2017). We concatenate the probabilities produced by base classifiers and feed them into SVM to get the final label.

3.3 Data

The training, development and test data all come from two sources: Chen et al. (1996) for news text in traditional Chinese and McEnery and Xiao (2003) for simplified Chinese. There are 18,769 sentences for training, 2,000 sentences for development, and 2,000 sentences for test. The datasets are balanced across two classes.

4 Results and Discussion

We report the results of each base classifier and the ensemble models on both the development set and the test set.

4.1 Results on Dev Set

The results for development set are presented in Table 1. Most of the base classifiers achieve an F-measure between 0.87 and 0.90. In particular, a word n-gram Naive Bayes (NB) model can reach 0.88 F-score. Using syntactic features (SYN) produces slightly worse results, likely due to the sentence-based nature of the task. That is, the syntactic features are very sparse, unlike in related tasks where the classification is on the document level (Bykh and Meurers, 2014; Hu et al., 2018). Models based on word level neural networks (LSTM and SEQ) have similar results, while adding pre-trained word embeddings improves the results (EMB). BERT with pre-trained character embeddings gives comparable results as EMB.

The best results are achieved using our ensemble models described in section 3.2. They are able to improve the F score by 0.01-0.02 over the best preforming base classifier, illustrating the robustness of ensemble models.

4.2 Results on Test Set

For the campaign, we submit the predictions of the three ensemble models. The results are presented in Table 2. The MaxProb and SVM ensemble models have similar performance, with an accuracy around 0.92. Our system is ranked as the first place for the simplified Chinese track and second place for the traditional Chinese track.

The confusion matrix of our best performing models are shown in Figure 1 and Figure 2. There seems to be a bias of predicting Taiwan sentences as Mainland sentences, the reason for which calls for further exploration.

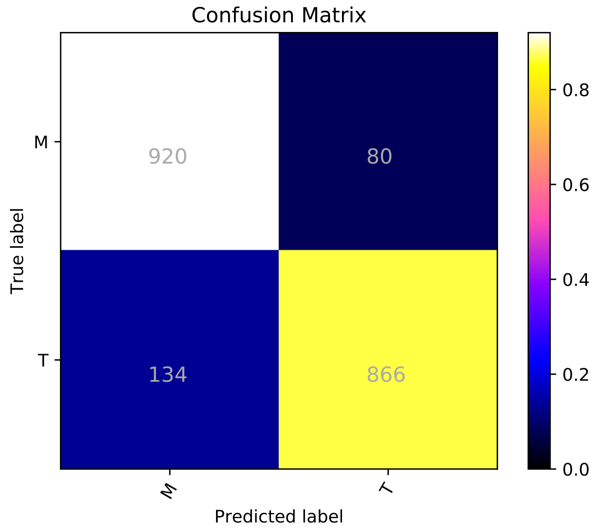

Figure 1: Sub-task DMT-simplified, Ensemble SVM

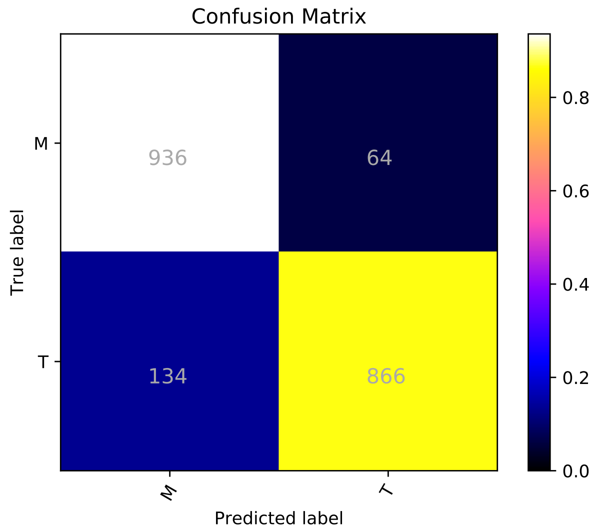

Figure 2: Sub-task DMT-traditional, Ensemble MaxProb

4.3 Prominent Features in Mainland and Taiwan Text

Now we examine word unigram features of the classes more closely. Using Information Gain described in Liu et al. (2016), we rank all the word unigram features. The top 20 features are shown in Table 3. First we notice proper nouns that are understandably distinctive, e.g., the word "Taiwan" appears more frequent in Taiwan news, while "Shenzhen"–a city in Mainland China–shows up only in Mainland news. Some other words have to do with the political system in Mainland, e.g., "comrade" and "socialism" occur only in Mainland news.

	BERT	LSTM	SEQ	EMB	NB	SYN	Ensemble MaxProb	Ensemble MaxMeanProb	Ensemble SVM
Simp.	**0.900**	0.874	0.878	0.893	0.881	0.854	0.917	0.910	0.921
Trad.	0.905	0.879	0.891	**0.907**	0.888	0.884	0.924	0.920	0.934

Table 1: Macro-averaged F score on **development** set for simplified and traditional Chinese. BERT: classifier using BERT pre-trained Chinese model. LSTM: word-level LSTM. SEQ: word-level sequential model classifier. EMB: neural network using pre-trained Chinese word embeddings. NB: word-level Naive-Bayes model. SYN: SVM with syntactic features.

System	**Simplified**	**Traditional**
Ensemble MaxProb	0.892	**0.901**
Ensemble MaxMeanProb	0.872	0.878
Ensemble SVM	**0.893**	0.899

Table 2: Macro-averaged F score on test set for simplified and traditional Chinese using ensemble models.

rank	ig_value	word	translation	freq Mainland	freq Taiwan
1	0.00237293	一个	one	290	0
2	0.00122218	学生	student	28	234
3	0.00118379	经济	economy	254	29
4	0.00094773	这个	this	116	0
5	0.00092902	台湾	Taiwan	19	172
6	0.00090684	全国	whole country	111	0
7	0.00079236	同志	comrade	97	0
8	0.0006871	网路	internet	0	77
9	0.00065798	我们	we	325	104
10	0.00061201	就是	be	82	1
11	0.00060674	资讯	information	0	68
12	0.00060137	改革	reform	102	6
13	0.00059619	深圳	Shenzhen (city)	73	0
14	0.00059442	使用	use	11	107
15	0.00058054	人们	people	106	8
16	0.00057632	记者	journalists	142	21
17	0.00054397	企业	enterprises	232	66
18	0.00053899	社会主义	socialism	66	0
19	0.00052829	人民	masses	127	18
20	0.00052387	为了	in order to	71	1

Table 3: Top 20 features selected by information gain

However, the top ranking feature 一个 (one) appears to be a segmentation error. It turns out that it is segmented as two words in Taiwan news, i.e., 一|个, but it is one word in Mainland news. In fact, it appears 143 times in the Taiwan training data (still half the frequency of Mainland news). The same goes for 这个 (this) and 全国 (whole country), which is segmented as two words in Taiwan news, but one in Mainland news.

Perhaps the only interesting lexical variations between Mainland and Taiwan in the top 20 features are 网路 (internet) and 资讯 (information), which are immediately recognized by the authors as Taiwan Mandarin. In Mainland China, they would be 网络 (internet) and 信息 (information).

5 Conclusion

In this paper we have described the IUCL ensemble models that distinguish Mainland and Taiwan Mandarin Chinese. We show that neural networks using pre-trained character/word embeddings outperform traditional *n*-gram models, and ensemble models can further improve the results over base classifiers. Although neural networks produce strong empirical results, traditional classifiers like SVM still play an important role when we need to investigate the importance of different features.

6 Acknowledgement

He Zhou and Hai Hu are funded by China Scholarship Council.

References

Harald Baayen, Hans Van Halteren, and Fiona Tweedie. 1996. Outside the cave of shadows: Using syntactic annotation to enhance authorship attribution. *Literary and Linguistic Computing*, 11(3):121–132.

Serhiy Bykh and Detmar Meurers. 2014. Exploring syntactic features for native language identification: A variationist perspective on feature encoding and ensemble optimization. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pages 1962–1973.

Keh-Jiann Chen, Chu-Ren Huang, Li-Ping Chang, and Hui-Li Hsu. 1996. SINICA CORPUS : Design Methodology for Balanced Corpora. In *Language, Information and Computation : Selected Papers from the 11th Pacific Asia Conference on Language, Information and Computation : 20-22 December 1996, Seoul*, pages 167–176, Seoul, Korea. Kyung Hee University.

François Chollet et al. 2015. Keras. https://keras.io.

Ronan Collobert and Jason Weston. 2008. A unified architecture for natural language processing: Deep neural networks with multitask learning. In *Proceedings of the 25th international conference on Machine learning*, pages 160–167. ACM.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Michael Gamon. 2004. Linguistic correlates of style: authorship classification with deep linguistic analysis features. In *Proceedings of the 20th international conference on Computational Linguistics*, page 611. Association for Computational Linguistics.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation*, 9(8):1735–1780.

Hai Hu, Wen Li, and Sandra Kübler. 2018. Detecting syntactic features of translated chinese. In *Proceedings of the Second Workshop on Stylistic Variation*, pages 20–28.

Chu-Ren Huang. 2009. Tagged chinese gigaword version 2.0, ldc2009t14. *Linguistic Data Consortium*.

Chu-Ren Huang and Lung-Hao Lee. 2008. Contrastive approach towards text source classification based on top-bag-of-word similarity. In *Proceedings of the 22nd Pacific Asia Conference on Language, Information and Computation*.

Tommi Jauhiainen, Marco Lui, Marcos Zampieri, Timothy Baldwin, and Krister Lindén. 2018. Automatic language identification in texts: A survey. *arXiv preprint arXiv:1804.08186*.

Dan Jurafsky and James H Martin. 2014. *Speech and language processing*, volume 3. Pearson London.

Sang-Bum Kim, Kyoung-Soo Han, Hae-Chang Rim, and Sung Hyon Myaeng. 2006. Some effective techniques for naive bayes text classification. *IEEE transactions on knowledge and data engineering*, 18(11):1457–1466.

Shen Li, Zhe Zhao, Renfen Hu, Wensi Li, Tao Liu, and Xiaoyong Du. 2018. Analogical reasoning on chinese morphological and semantic relations. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 138–143. Association for Computational Linguistics.

Wen Li and Liang Zou. 2017. Classifier stacking for native language identification. In *Proceedings of the 12th Workshop on Innovative Use of NLP for Building Educational Applications*, pages 390–397.

Can Liu, Wen Li, Bradford Demarest, Yue Chen, Sara Couture, Daniel Dakota, Nikita Haduong, Noah Kaufman, Andrew Lamont, Manan Pancholi, et al. 2016. Iucl at semeval-2016 task 6: An ensemble model for stance detection in twitter. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, pages 394–400.

Shervin Malmasi, Marcos Zampieri, Nikola Ljubešić, Preslav Nakov, Ahmed Ali, and Jörg Tiedemann. 2016. Discriminating between similar languages and arabic dialect identification: A report on the third dsl shared task. In *Proceedings of the 3rd Workshop on Language Technology for Closely Related Languages, Varieties and Dialects (VarDial)*, Osaka, Japan.

Christopher Manning, Mihai Surdeanu, John Bauer, Jenny Finkel, Steven Bethard, and David McClosky. 2014. The stanford corenlp natural language processing toolkit. In *Proceedings of 52nd annual meeting of the association for computational linguistics: system demonstrations*, pages 55–60.

A. M. McEnery and R. Z. Xiao. 2003. The lancaster corpus of mandarin chinese.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119.

Vivek Narayanan, Ishan Arora, and Arjun Bhatia. 2013. Fast and accurate sentiment classification using an enhanced naive bayes model. In *International Conference on Intelligent Data Engineering and Automated Learning*, pages 194–201. Springer.

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.

DIAO Yan-bin. 2012. The situation and thinking about the comparative study of language in the four places across the taiwan strait [j]. *Chinese Language Learning*, 3.

Marcos Zampieri, Shervin Malmasi, Nikola Ljubešić, Preslav Nakov, Ahmed Ali, Jörg Tiedemann, Yves Scherrer, and Noëmi Aepli. 2017. Findings of the VarDial Evaluation Campaign 2017. In *Proceedings of the Fourth Workshop on NLP for Similar Languages, Varieties and Dialects (VarDial)*, Valencia, Spain.

Marcos Zampieri, Shervin Malmasi, Preslav Nakov, Ahmed Ali, Suwon Shuon, James Glass, Yves Scherrer, Tanja Samardžić, Nikola Ljubešić, Jörg Tiedemann, Chris van der Lee, Stefan Grondelaers, Nelleke Oostdijk, Antal van den Bosch, Ritesh Kumar, Bornini Lahiri, and Mayank Jain. 2018. Language Identification and Morphosyntactic Tagging: The Second VarDial Evaluation Campaign. In *Proceedings of the Fifth Workshop on NLP for Similar Languages, Varieties and Dialects (VarDial)*, Santa Fe, USA.

Marcos Zampieri, Shervin Malmasi, Yves Scherrer, Tanja Samardžić, Francis Tyers, Miikka Silfverberg, Natalia Klyueva, Tung-Le Pan, Chu-Ren Huang, Radu Tudor Ionescu, Andrei Butnaru, and Tommi Jauhiainen. 2019. A Report on the Third VarDial Evaluation Campaign. In *Proceedings of the Sixth Workshop on NLP for Similar Languages, Varieties and Dialects (VarDial)*. Association for Computational Linguistics.

Marcos Zampieri, Liling Tan, Nikola Ljubešić, Jörg Tiedemann, and Preslav Nakov. 2015. Overview of the dsl shared task 2015. In *Proceedings of the Joint Workshop on Language Technology for Closely Related Languages, Varieties and Dialects (LT4VarDial)*, pages 1–9, Hissar, Bulgaria.