Grammatical Error Detection Based on Machine Learning for Mandarin as Second Language Learning

Jui-Feng Yeh*
Department of Computer and information Science
National Chia-Yi University
Chiayi, Taiwan (R.O.C.)
raiph@mail.ncyu.edu.tw

Tsung-Wei Hsu
Department of Computer and information Science
National Chia-Yi University
Chiayi, Taiwan (R.O.C.)
s1050480@mail.ncyu.edu.tw

Chan-Kun Yeh
Department of Computer and information Science
National Chia-Yi University
Chiayi, Taiwan (R.O.C.)
s1030484@mail.ncyu.edu.tw

Abstract

Mandarin is not simple language for foreigner. Even using Mandarin as the mother tongue, they have to spend more time to learn when they were child. The following issues are the reason why causes learning problem. First, the word is envolved by Hieroglyphic. So a character can express meanings independently, but become a word has another semantic. Second, the Mandarin's grammars have flexible rule and special usage. Therefore, the common grammatical errors can classify to missing, redundant, selection and disorder. In this paper, we proposed the structure of the Recurrent Neural Networks using Long Short-term memory (RNN-LSTM). It can detect the error type from the foreign learner writing. The features based on the word vector and part-of-speech vector. In the test data found that our method in the detection level of recall better than the others, even as high as 0.9755. That is because we give the possibility of greater choice in detecting errors.

1 Introduction

In recent years, the rapid development of communication between countries. Especially the Chinese region, more and more foreign people came to traveling or working. So the Mandarin become the option as second language learner. But it is not easy to learn because its grammars are very complexity.

To research Mandarin as second language, we can distinguish two parts: word level and sentence level. In word level, there have two main aspects are Word Segmentation and Part-of-Speech (POS) Tagging. We want to segment the sentence to the basic semantic units and give the correct tagging. About the research of word segmentation and POS tagging, (Ye, J. et al., 2011) the authors proposed using the prefix and suffix query of Chinese word segmentation algorithm for maximum matching. This structure can choose the best structure of words as the dictionary. (Li, Zhenghua et al., 2014) the authors proposed joint algorithm to optimize the POS tagging and dependency parsing. They use the parsing tree to find the relationship between words and sentence. (Ma, Wei-Yun and Chen, Keh-Jiann, 2005) the authors proposed the system to word segmentation and POS tagging about Chinese. They define the 47 class of POS in Chinese and this system is now using in Taiwan Academia Sinica. And we employ this POS classification in our research.

In the word level, the Chinese common grammar error can classify the four parts: Missing, Redundant, Selection, Disorder (see example in Table 1). In the grammar and word order, (Xiao Sun and Xiaoli Nan, 2010) proposed using latent semi-CRF model on the Chinese phrase classifications. (Jinjin Zhu and Yangsen Zhang, 2010) the authors proposed auto-detect the Chinese errors by using hybrid algorithm. They are looking for word, syntax and semantic. (B. Zhang et al., 2010) the authors proposed extracting opinion sentence by SVM and syntax template. Then in the grammar error detection, (H. H. Feng et al., 2016) the authors proposed Automated Error Detection of ESL (English as a Second Language) Learners. And (Chung-Hsien Wu et al., 2010) the authors proposed sentence correction incorporating relative position and parse template language models. They are looking for the English errors. Then in Chinese error detection, (Lung-Hao Lee et al., 2013) proposed the linguistic rules of Chinese error detection for CFL (Chinese as a For-

Proceedings of the 3rd Workshop on Natural Language Processing Techniques for Educational Applications,
pages 140–147, Osaka, Japan, December 12 2016.

eign Language). And (Chi-Hsin Yu et al., 2012) the authors proposed detect the errors of word order by training the HSK corpus. The HSK corpus is simplified Chinese data. Then (Shuk-Man Cheng et al., 2014) they also using HSK corpus to proposed word ordering errors detection and correction by SVM to ranking the optimal sentences.

Table 1: Common grammatical error type

Grammatical error types	**Examples of erroneous sentences**	**Examples of correct sentences**
漏字錯誤(Missing)	我送你家 (I take you home.)	我送你**回**家
冗詞錯誤(Redundant)	他是我**的**最重要的朋友 (He is my important friend.)	他是我最重要的朋友
詞彙誤用(Selection)	我是騎腳踏車的**拿手**	我是騎腳踏車的**好手**
語序運用不當(Disorder)	我**去學校早上** (I go to school in the morning.)	我**早上去學校**

In our research, we proposed the architecture for grammatical error detection by recurrent neural network using long-short term memory (RNN-LSTM) as a second language learner. We use this architecture to generate the language model and error rule patterns are made based on parsing tree.

2 Method

In this section, The processing flow is illustrated here. There are distinguish two phases: training phase and testing phase. In training phase, we were doing word segmentation and part-of-speech (POS) by CKIP (Chinese Knowledge and Information Processing) Autotag. Then classify words to several class and transform the sentence to the word vector. We will explain how to classify words in section 2.1. And we will describe how to generate the language model by RNN-LSTM in the section 2.2. Final, we show some parsing tree examples to explain the error pattern model in the section 2.3. In section 2.4, we explain the testing phase in our system how to detect the grammatical error.

2.1 Word Clustering

How to express the meaning of a word in the computer? In traditional methods, we could research the semantic dictionary. Such as WordNet for English or E-HowNet for Chinese. They have to spend a lot of time to tagging by people.

In our method to clustering word is based on probability from (Franz J. O., 1999) proposed model. $P(w_1^N)$ represent a sentence sequence. $w_1^N = w_1 ..... w_N$ represent the set of the words. The probability of the context of words in sentence is

$$P(w_1^N) = \prod_{t=1}^{N} p(w_t \mid w_{t-1}) \quad (1)$$

We made the close probability of the words to C classes. So we can represent the relationship of sentence correspond to classes:

$$P(w_1^N \mid C) = \prod_{t=1}^{N} p(w_t \mid C(w_t)) \cdot p(C(w_t) \mid C(w_{t-1})) \quad (2)$$

Where $P(w_t \mid C(w_t))$ represent the relationship of words correspond to classes. $P(C(w_t) \mid C(w_{t-1}))$ represent the relationship of context of classes.

Then choose the best classification from all class. It can represent by

$$\hat{C} = \underset{c}{\operatorname{argmax}}\ p(w_1^N \mid C) \tag{3}$$

If the word's probability is use formula (2) and combine the maximum likelihood algorithm from (Kneser, 1999) proposed to get optimal likelihood. It can represent by

$$\mathrm{ML}(C,n) = -\sum_{C,C'} n(C,C') \ln n(C,C') + 2\sum_{C} n(C) \ln(C) \tag{4}$$

$$\hat{C} = \underset{c}{\operatorname{argmax}}\ \mathrm{ML}(C,\ n) \tag{5}$$

Where n(.) represent the probability in the training corpus. In this paper, we use the classification model to classify the words in the training corpus and build the codebook for query.

2.2 RNN-LSTM

In this section, the depth of learning architecture and why the use of recurrent neural network (RNN) to training model. And analyse the sentence structure with the concept of the parsing tree. Then replace hidden units to long-short term memory (LSTM) units in RNN hidden layer. RNN's horizontal nodes of the hidden layer are connected. So this structure suitable for train the length of different sentences with represent the contextual relationship.

The Figure 1 is the structure of RNN and it has three parts: input layer, hidden layer, and output layer. The hidden layer can have many layer in this structure so we assume the 30 layer in hidden layer to train the optimum parameters. And it shows that the training process is carried out by x_0 to x_t. So the cost function in the time t is

$$J = -y_t \log(n_t) - (1 - y_t) \log(1 - n_t) \tag{6}$$

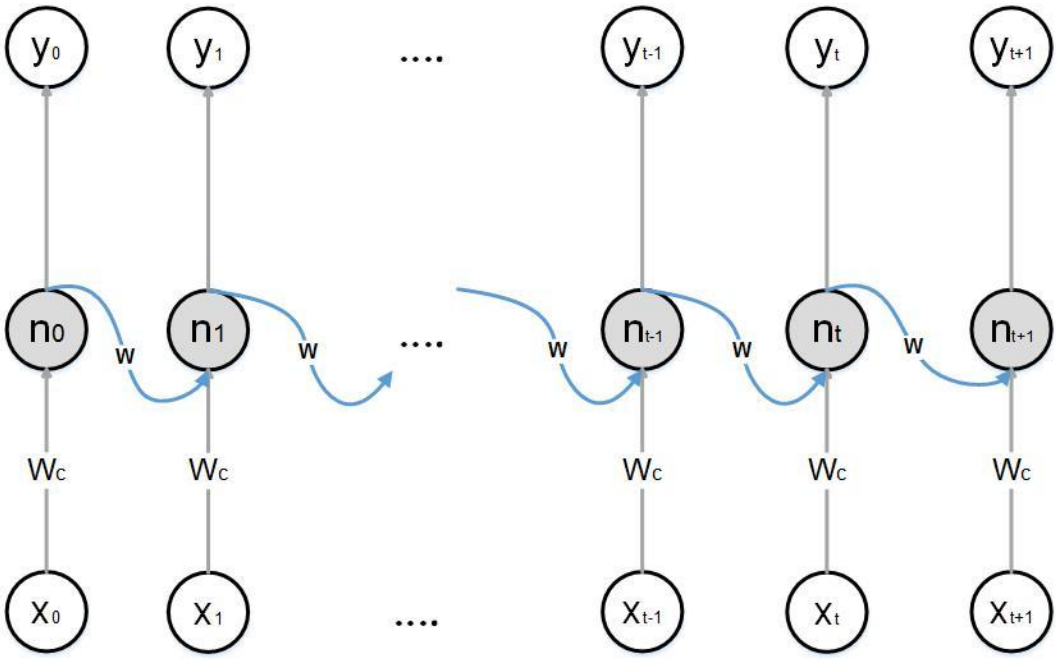


Figure 1: The sentence input in RNN-LSTM

The figure 2 shows the RNN traditional unit and LSTM unit in the hidden layer. In figure 2(a), we could find the unit input then using the sigmoid function to normalize. The sigmoid is shown $\sigma(x) = 1/1 + \exp(-x)$. So the hidden unit n_t is

$$n_t = \sigma(\omega_c x_t + \omega_p n_{t-1}) \tag{7}$$

Where ω_c and ω_p is the weights of the current input and previous output. And the output y_t is

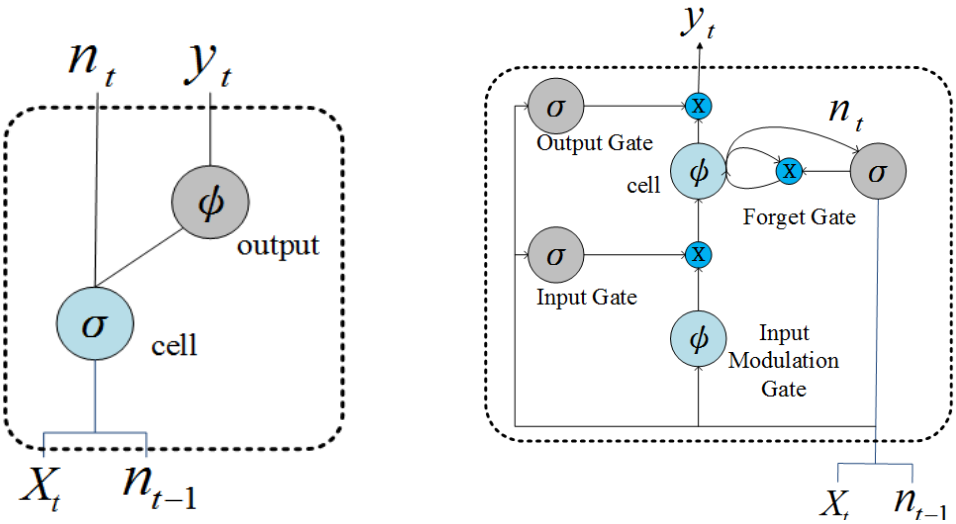
$$y_t = \varphi(\omega_t x_t) \tag{8}$$

In figure 2(b), we could see the three gates in the LSTM unit: input gate, forget gate, and output gate. First, the input gate IG controlled whether cells in the input layer can enter. Second, the forget gate FG controlled whether cells in the hidden layer can enter and output to next node. Final, the output gate OG controlled the current cell output. Then the formula (9) ~ (11) represent IG, FG, OG:

$$\mathrm{IG} = \sigma(\omega_{\mathrm{i}} x_t + \omega_p x_{\mathrm{t\text{-}1}}) \tag{9}$$

$$\mathrm{FG} = \sigma(\omega_{\mathrm{c}} x_t + \omega_p \mathrm{n}_{\mathrm{t\text{-}1}}) \tag{10}$$

$$\mathrm{OG} = \sigma(\omega_{\mathrm{o}} x_t + \omega_p x_{\mathrm{t\text{-}1}}) \tag{11}$$



(a) (b)

Figure 2: (a) RNN traditional unit (b) LSTM unit

Where we can using (9)~(11) to get the cell result in hidden layer and the output y_t are

$$cell = FG + IG \tanh(\omega_c x_t + \omega_p n_{t-1}) \tag{12}$$

$$y_t = OG \tanh(cell) \tag{13}$$

2.3 Error Pattern Model

In this paper, we focus on the English-speaking learners who are influenced by their native language and build the error pattern model. First, we need to analysis the error pattern in the sentence which using the parsing tree. We integrated the concept of RNN-LSTM to detect the error patterns. The bottom node of parsing tree is the input node in RNN-LSTM. The parent node is the LSTM node.

2.4 Testing Phase

First, read the test data and then word segmentation and part-of-speech tagging. Then the sentence according to the codebook to vector expression, and input the training model. We could get the probabilities from every types' model. After get the probabilities, we compare four errors with the correct probability to find all possible errors. And the output format is <sentence id, start position, end position, error type>. There is not only one error type in a sentence; it maybe has two or more errors. In addition to the system detected, we also adopted the error pattern model as the final output.

3 Experiment

In this section, we analyse the performance of the proposed architecture. First, we introduced the corpus in our training model and the evaluation of testing. Final, we showed the experiment result in training model and the result of NLP-TEA 3 competition.

3.1 Data and Evaluation Criterion

We used the three datasets from NLP-TEA 1(Yu, Liang-Chih et al, 2014) to NLP-TEA 3. There are two datasets: TOCFL corpus (Traditional Chinese) and HSK corpus (Simplified Chinese), the details are showed in the table 2.

Table 2: The Training Corpus

SOURCE / SENTENCES	Missing	Redundant	Selection	Disorder	Correct
TOCFL	6328	4122	5439	1621	18483
HSK	2810	2322	3834	896	10071

In this paper, we have two evaluation criterion: perplexity (Oparin et al, 2012) and confusion matrix. Perplexity is used to evaluate the performance of language model training from RNN-LSTM. Its format can represent:

$$\mathrm{PPL}_n = \exp(-\frac{1}{I_n}\sum_{i \in I_n} \log p(w_i \mid w_1^{i-1})) \tag{14}$$

In addition, we used three parameters based on the confusion matrix to evaluate our system. They are precision, recall, and F1-score and can represented:

$$precision = \frac{tp}{tp + fp} \tag{15}$$

$$recall = \frac{tp}{tp + fn} \tag{16}$$

$$F1 - Score = 2 * \frac{precision * recall}{precision + recall} \tag{17}$$

3.2 Experiment Result

First, we wanted to find the optimal class to our language model in the training phase. Therefore, we used the perplexity to evaluate and showed the result in table 3. In the table, we could see the 30-class is in average better than other classes. And we use internal validation and proved the 30-class is better.

Table 3: The Perplexity of language model to each type

	30 class	35 class	40 class	45 class
Missing	**167.5952**	183.9607	226.9839	179.2754
Redundant	**178.8971**	217.7797	209.461	179.3632
Selection	188.2969	206.3802	242.5115	**156.4807**
Disorder	250.8187	282.3815	262.3684	**248.5769**
Correct	130.5262	121.8946	**85.0405**	101.9611

Therefore, we chose the 30-class to training and used to the test phase. Second, we showed the result from NLP-TEA 2016.

In detection level (see the Table 4), our recall is better than other teams. It means we can find more error rate in dataset. In addition, our F1-Score is the best in this level. It means our overall is superior to the others, although our precision is less than other teams.

In identification level (see the Table 5), it show who can find most error and error type is correct. In our method, we found that our recall is better than other teams. It means we find more correct error type than other teams, although our precision is less than other teams. Nevertheless, our F1-Score is better than NCTU+NTUT.

In Position level (Table 5), our method that looking for accurate location is not illustrious in this level. We consider the reasons are our correction is not enough standard.

Table 4, Table 5, and Table 6 are the performance with the NLP-TEA 2016 TOCFL dataset and compare the others team

Table 4: Detection level

	Accuracy	**Precision**	**Recall**	**F1**
NCYU	0.5218	0.5202	**0.9726**	**0.6779**
NCTU+NTUT	0.5442	**0.6593**	0.246	0.3583
CYUT	**0.5955**	0.6259	0.5419	0.5809

Table 5: Identification-level

	Accuracy	**Precision**	**Recall**	**F1**
NCYU	0.2328	0.2265	**0.4744**	0.3066
NCTU+NTUT	0.511	0**.4892**	0.1224	0.1958
CYUT	**0.5154**	0.46	0.3021	0.3647

Table 6: Position-level

	Accuracy	**Precision**	**Recall**	**F1**
NCYU	0.0231	0.0129	0.0195	0.0155
NCTU+NTUT	**0.4603**	**0.2542**	0.0483	0.0811
CYUT	0.3113	0.1461	0.1089	**0.1248**

In detection level (see the Table 7), our recall is better than other teams. It means we can find more error rate in dataset. Although our precision is less than other teams, our F1-Score is better than SKY's method.

In Identification level (see the Table 8), our recall is better than SKY's method that we can find more correct error type. However, our precision is less than other teams.

In Position level (see the Table 9), our method that looking for accurate location is not illustrious in this level. We consider the reasons are our correction is not enough standard.

Table 7, Table 8, and Table 9 are the performance with the NLP-TEA 2016 HSK dataset and compare the others team

Table 7: Detection level

	Accuracy	Precision	Recall	F1
NCYU	0.5042	0.4964	**0.9755**	0.658
HIT	0.637	0.6071	0.7296	**0.6628**
SKY	0.6579	**0.8746**	0.3505	0.5005

Table 8: Identification-level

	Accuracy	Precision	Recall	F1
NCYU	0.2687	0.2588	0.5263	0.347
HIT	0.5565	0.5002	**0.5447**	**0.5215**
SKY	0.6765	**0.8821**	0.2972	0.4446

Table 9: Position-level

	Accuracy	Precision	Recall	F1
NCYU	0.0312	0.0158	0.0217	0.0183
HIT	0.4475	0.3695	**0.3697**	0.3696
SKY	0.6376	0.7054	0.2217	0.3373

Conclusion

In this paper, we present a method using conditional random field model for predicting the grammatical error diagnosis for learning Chinese. In the grammatical error diagnosis, not only do we find a single error, but we can also find a sentence with multiple errors. After observe the experiment results, our method is acceptable in NLP-TEA 2016. We believe this system is feasible. This system is useful for a foreign who learn Chinese as a second language. Even the people who use Chinese as a first language might use the wrong grammars. There are some issues should be revise. First, finding the best way to solve the problem to find the precise location. Second, increase the ranking mechanism to find the optimal words to correct the sentence. In the future, we will pay attention to improve the precision and recall rates in this system. Let it can automatic correct the error if the people input the sentences.

Reference

Ye, J., Li, S., Hao, G., Li, S., Yang, Y., & Jin, C. (2011, October). The prefix and suffix query of Chinese word segmentation algorithm for maximum matching. In 2011 International Conference on Image Analysis and Signal Processing (pp. 74-77). IEEE.

Li, Z., Zhang, M., Che, W., Liu, T., & Chen, W. (2014). Joint Optimization for Chinese POS Tagging and Dependency Parsing. Audio, Speech, and Language Processing, IEEE/ACM Transactions on, 22(1), 274-286.

Ma, W. Y., & Chen, K. J. (2005). Design of CKIP Chinese word segmentation system. Chinese and Oriental Languages Information Processing Society, 14(3), 235-249.

X. Sun, & X. Nan, "Chinese base phrases chunking based on latent semi-CRF model," In International Conference on Natural Language Processing and Knowledge Engineering (NLP-KE), pp. 1-7, IEEE, August, 2010.

Z. Jinjin, & Z. Yangsen, "Research and implementation on a hybrid algorithm for Chinese automatic error-detecting," In International Conference on Artificial Intelligence and Computational Intelligence (AICI), vol. 1, pp. 413-417, IEEE, October, 2010.

B. Zhang, Y. Zhou, & Y. Mao, “Extracting opinion sentence by combination of SVM and syntactic templates,” In International Conference on Natural Language Processing and Knowledge Engineering (NLP-KE), pp. 1-7, IEEE, August, 2010.

H. H. Feng, A. Saricaoglu, & E. Chukharev-Hudilainen, “Automated Error Detection for Developing Grammar Proficiency of ESL Learners,” calico journal, vol. 33, no. 1, pp. 49, 2016.

C. H. Wu, C. H. Liu, M. Harris, & L. C. Yu, “Sentence correction incorporating relative position and parse template language models,” IEEE Transactions on Audio, Speech, and Language Processing, vol. 18, no. 6, pp. 1170-1181, 2010.

L. H. Lee, L. P. Chang, K. C. Lee, Y. H. Tseng, & H. H. Chen, “Linguistic rules based Chinese error detection for second language learning,” In Work-in-Progress Poster Proceedings of the 21st International Conference on Computers in Education (ICCE-13), pp. 27-29, November, 2013.

C. H. Yu, & H. H. Chen, “Detecting Word Ordering Errors in Chinese Sentences for Learning Chinese as a Foreign Language,” In COLING, pp. 3003-3018, 2012.

Shuk-Man Cheng, Chi-Hsin Yu, and Hsin-Hsi Chen, "Chinese Word Ordering Errors Detection and Correction for Non-Native Chinese Language Learners," Proceedings of COLING’14, pp. 279-289, 2014.

Och, F. J. (1999, June). An efficient method for determining bilingual word classes. In Proceedings of the ninth conference on European chapter of the Association for Computational Linguistics (pp. 71-76). Association for Computational Linguistics.

Kneser, R., & Ney, H. (1993, September). Improved clustering techniques for class-based statistical language modelling. In Eurospeech (Vol. 93, pp. 973-76).

Agrawal, R., & Srikant, R. (1994, September). Fast algorithms for mining association rules. In Proc. 20th int. conf. very large data bases, VLDB (Vol. 1215, pp. 487-499).

Yu, Liang-Chih, Lung-Hao Lee, and Li-Ping Chang. 2014. Overview of grammatical error diagnosis for learning Chinese as a foreign language. In Proceedings of the 1st Workshop on Natural Language Processing Techniques for Educational Applications (NLP-TEA 2014). 42-47

Lee, Lung-Hao, Liang-Chih Yu, and Li-Ping Chang. 2015. Overview of the NLP-TEA 2015 shared task for Chinese grammatical error diagnosis. In Proceedings of the 2nd Workshop on Natural Language Processing Techniques for Educational Applications (NLP-TEA 2015). 1-6.

Oparin, I., Sundermeyer, M., Ney, H., & Gauvain, J. L. (2012, March). Performance analysis of neural networks in combination with n-gram language models. In 2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) (pp. 5005-5008). IEEE.