Directed Reading

Chinese Spelling Check

Hantao Li
G2101725H
HLI038@e.ntu.edu.sg

ABSTRACT

Grammatical Error Correction (GEC) refers to the automatic detection of sentence grammatical errors and then correcting the detected errors to reduce the cost of manual verification. **Chinese Spelling Check (CSC)** is a sub-task of grammatical error correction, which detects and corrects spelling errors in the Chinese nature language. In this directed reading report, we pick up three papers from ACL in recent two years which focus on the topic of CSC. We have also read several other articles about CSC, with only a general understanding of its methods, not as carefully as the above three papers. Moreover, this report will not only contain the analysis of the above papers but also a brief view on the development of the CSC field nowadays.

1 Introduction

1.1 Background

People usually produce grammatical errors due to the randomness of the text input and the lack of subsequent review. In recent years, with the upsurge of self-media, everyone could become a producer of information; thus, the grammatical errors on the Internet have increased sharply, which significantly impact the user experience. Therefore, the research of grammatical error correction (GEC) emerges as the times require. It is a technology to detect typos in a text and correct them. Although GEC has essential applications in many practical scenarios, this task has always been tepid in the academic field. In recent years, the research results have been much less than machine translation and information extraction.

English is composed of 26 letters; thus, it may not be in the vocabulary when misspelling an English word. However, Chinese is composed of thousands of commonly used characters. Limited by the 'pinyin' or handwriting input method, similar characters are easily misused. Therefore, there is a particular task in Chinese grammar error correction: Chinese spelling check (CSC). In particular, the input and output of the typo task are entirely aligned, that is, changing one character to another.

In CSC, an essential part of implementation is tokenization. Consistent with the knowledge mentioned in the course slides, research is more based on character models than word models since it uses many sub-modules and deals with many special cases independently, resulting in high system complexity and complicated global optimization. Moreover, on the one hand, the error in Chinese is usually because of the misusing of characters due to their similarity. According to Liu et al. [1], Most errors in Chinese, about 83%, are due to phonological similarity, while the visual similarity is responsible for 48% of errors. On the other hand, there are two types of misusing: one is semantic errors, such as utilization errors and context collocation errors, and the other is intellectual errors, which means lacking world knowledge, such as proper nouns. We can see that the academic field mainly focuses on semantic errors, while intellectual errors are generally solved when they are realized by the industry.

Finally, GEC has strict requirements for the misjudgment rate, which generally must be less than 0.5%. If the error rate of the error correction method is very high (FP for the correction model, which means correct the correct character into wrong), it will have a significant adverse effect on the system and user.

1.2 Methods

CSC has been studied for many years. The standard methods can be summarized as establishing dictionaries, editing distance, language model, etc.

The construction of a dictionary has a tremendous cost and is only suitable for some specific fields with limited errors. Editing distance adopts a method similar to string fuzzy matching, which can correct errors by comparing with the correct samples; however, it is not universal.

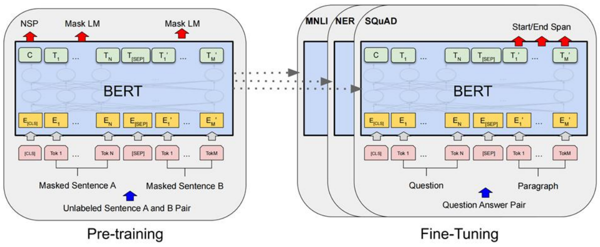


Figure 1: The Architecture of BERT. [2]

Therefore, the focus point of academic and industrial research is based on the language model (LM). Before 2018, the language model method could be divided into traditional n-gram LM and DNN LM.

After 2018, the pre-training language model became popular. Researchers quickly migrated the BERT model, which was proposed by Google, to GEC and achieved new optimal results. The main difference between BERT and the previous deep learning models is that, in the pre-training stage, the two tasks of 'Masked LM' (MLM) and 'Next Sentence Prediction' (NSP) are used, as shown in the left part of Figure 1. [2] The three papers we picked up are all related to BERT, which improved and optimized the CSC methods based on BERT.

2 Directed Reading

2.1 Soft-Masked BERT [3]

To maximize the efficacy of BERT, ByteDance AI Lab and Fudan University's researchers published the paper on ACL2020: ***Spelling Error Correction with Soft-Masked BERT***.

2.1.1 Motivation. The primary method of CSC based on BERT is to pre-train on the unlabeled data set, construct the training data by using the confusion set through data enhancement, and then predict the most likely character in the candidate for each position a given sentence. However, since BERT does not have enough ability to detect whether there are errors at each position, the accuracy of this method may not be optimal, which is due to the way of pre-training by using mask language modeling. Only 15% of the characters will be covered during pre-training, and the model will only learn the distribution of mask tokens, while other characters will not be corrected. Thus, the Recall of the model may not be enough.

2.1.2 Method. Consequently, the author of this paper proposed a novel neural network named Soft-Masked BERT, which contains two different networks. One is a detection network based on Bi-GRU, while another is a correction network based on BERT. The Architecture of Soft-Masked BERT is shown in Figure 2. The detection network based on Bi-GRU predicts the error probability of each character and then uses this probability to construct the soft masking embedding of each character. When the detection probability is 1, the character is equal to the mask vector, and when the probability is 0, the character is equal to the original character vector. The standard correction BERT model is connected later.

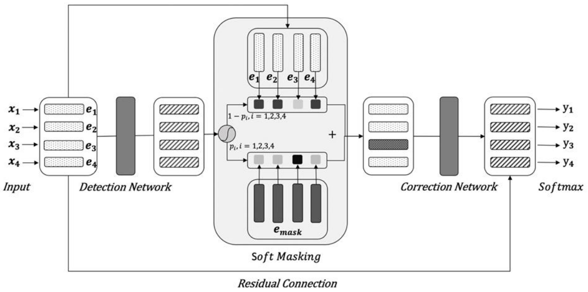


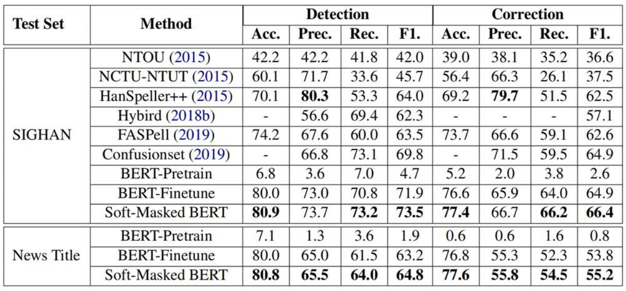
Figure 2: Architecture of Soft-Masked BERT. [3]

For the detection network and soft masking, the output soft-masked embedding e'_i for the *i*-th character is the adding result with the probability of masking character and the inverse probability of the original character. In other words, the smaller the probability of character error calculated by the detection network, the closer the result of soft masking output is to the original e_i in input embedding and vice versa.

For the correction network, it is a sequential multi-classification marker model based on BERT, which detects the characteristics of the network output as the input of the BERT 12-layer transformer module, and the output of the last layer adding with the residual connection of the input part ei as the final feature representation of each character. Eventually, each feature is passed through a SoftMax classifier, and the character with the most significant probability is output from the candidate character list, which is considered the correct character at each position.

For the learning procedure, the loss function of the network is a linear weighted combination with parameter λ of the detection network and the correction network. In the later section, the authors show that the highest F1 score is obtained when λ=0.8..

2.1.3 Result. The author did a comparative experiment on 'SIGHAN' and 'new title' data sets. Among them, 'SIGHAN' is an open-source Chinese text error correction data set in 2013, with a scale of about 1000. 'News title' is an error correction data set automatically constructed from today's headlines, with 5 million corpora.



Test Set	Method	Detection				Correction			
		Acc.	Prec.	Rec.	F1.	Acc.	Prec.	Rec.	F1.
SIGHAN	NTOU (2015)	42.2	42.2	41.8	42.0	39.0	38.1	35.2	36.6
	NCTU-NTUT (2015)	60.1	71.7	33.6	45.7	56.4	66.3	26.1	37.5
	HanSpeller++ (2015)	70.1	**80.3**	53.3	64.0	69.2	**79.7**	51.5	62.5
	Hybird (2018b)	-	56.6	69.4	62.3	-	-	-	57.1
	FASPell (2019)	74.2	67.6	60.0	63.5	73.7	66.6	59.1	62.6
	Confusionset (2019)	-	66.8	73.1	69.8	-	71.5	59.5	64.9
	BERT-Pretrain	6.8	3.6	7.0	4.7	5.2	2.0	3.8	2.6
	BERT-Finetune	80.0	73.0	70.8	71.9	76.6	65.9	64.0	64.9
	Soft-Masked BERT	**80.9**	73.7	**73.2**	**73.5**	**77.4**	66.7	**66.2**	**66.4**
News Title	BERT-Pretrain	7.1	1.3	3.6	1.9	0.6	0.6	1.6	0.8
	BERT-Finetune	80.0	65.0	61.5	63.2	76.8	55.3	52.3	53.8
	Soft-Masked BERT	**80.8**	**65.5**	**64.0**	**64.8**	**77.6**	**55.8**	**54.5**	**55.2**

Figure 3: The performance of Soft-Masked BERT (%). [3]

Figure 3 shows the result table of the comparative experiment, which indicates that the proposed model Soft-Masked BERT outperforms the baseline methods on both datasets.

In a word, this paper introduces significant noise to the detected error characters, hoping to learn the correct characters, while the slight noise is introduced to other correct characters detected, avoiding changing the original characters. The effect of the experimental results has been improved, but not obvious, which is only one percentage point higher than that of BERT Finetune.

2.2 SpellGCN [4]

To maximize the efficacy of utilizing the confusion set, Ant Financial Services Group's researchers published the paper on

ACL2020: ***SpellGCN: Incorporating Phonological and Visual Similarities into Language Models for Chinese Spelling Check***.

2.2.1 Motivation. The motivation of this paper is that the BERT model training of errors only uses the semantic similarity; however, the similarity information between characters contains not only the semantic information but also inter-relationship in terms of pronunciation and shape, which are necessary so that the model can learn to generate related answers. This paper proposes integrating this information into the model, trying to fuse both the symbolic space (phonological and visual similarity knowledge) and the semantic space (language semantic knowledge) into one model.

2.2.2 Method. During BERT's MLM training, the parameters of the classification layer use the parameters of the input layer character embedding. Thus, the critical point is integrating the information of phonological and visual similarity into the corresponding parameters in the classification layer.

The highlight of this work is that the confusion sets are made into a 'graph' and added to the embedding of characters. SpellGCN requires two similarity graphs Ap and AS, for pronunciation and shape similarities correspondingly. Each similarity graph is a matrix of $\mathbb{R}^{N\times N}$ with the *N* characters in the confusion set. If A_{ij}=1 in the matrix, it means that i-th and j-th characters are similar character pairs. The architecture of similarity graphs is shown in the right part of Figure 4.

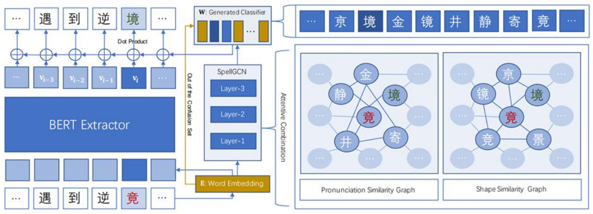
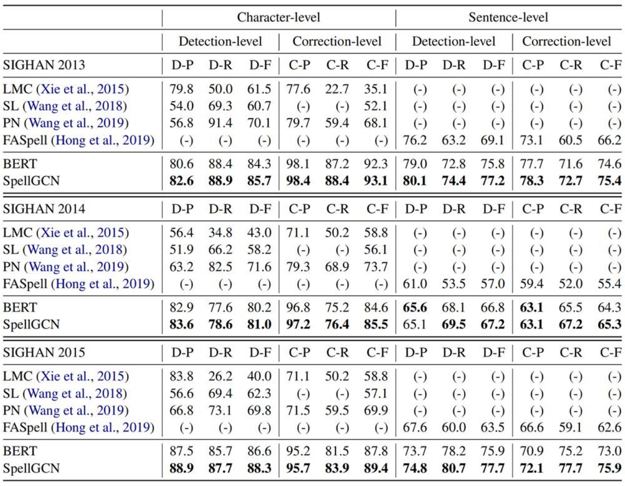


Figure 4: The framework of the proposed SpellGCN. [4]

The graph convolution operation is to operate on these two graphs and then use the attention mechanism to combine the representations of the two graphs. The final representation combines the representation of each layer in front with the final representation. Since the confusion set is only a part of all characters, the classification parameters corresponding to the thesaurus can be represented by GCN in the confusion set. For the rest characters, BERT-base is used to generate the input character embedding.

2.2.3 Result. The experiment was carried out on 'SIGHAN13', 'SIGHAN14' and 'SIGHAN15'. The experimental comparison was carried out in the two dimensions of character and sentence, respectively.

Figure 5 shows the result table of the comparative experiment, which indicates that the proposed model SpellGCN outperforms the baseline methods on both datasets. However, we must notice that the BERT model without any additional part has been outstanding; only 1-2 percentage points are added after GCN, showing no significant differences with the above paper.



	Character-level						Sentence-level					
	Detection-level			Correction-level			Detection-level			Correction-level		
SIGHAN 2013	D-P	D-R	D-F	C-P	C-R	C-F	D-P	D-R	D-F	C-P	C-R	C-F
LMC (Xie et al., 2015)	79.8	50.0	61.5	77.6	22.7	35.1	(-)	(-)	(-)	(-)	(-)	(-)
SL (Wang et al., 2018)	54.0	69.3	60.7	(-)	(-)	52.1	(-)	(-)	(-)	(-)	(-)	(-)
PN (Wang et al., 2019)	56.8	91.4	70.1	79.7	59.4	68.1	(-)	(-)	(-)	(-)	(-)	(-)
FASpell (Hong et al., 2019)	(-)	(-)	(-)	(-)	(-)	(-)	76.2	63.2	69.1	73.1	60.5	66.2
BERT	80.6	88.4	84.3	98.1	87.2	92.3	79.0	72.8	75.8	77.7	71.6	74.6
SpellGCN	**82.6**	**88.9**	**85.7**	**98.4**	**88.4**	**93.1**	**80.1**	**74.4**	**77.2**	**78.3**	**72.7**	**75.4**
SIGHAN 2014	D-P	D-R	D-F	C-P	C-R	C-F	D-P	D-R	D-F	C-P	C-R	C-F
LMC (Xie et al., 2015)	56.4	34.8	43.0	71.1	50.2	58.8	(-)	(-)	(-)	(-)	(-)	(-)
SL (Wang et al., 2018)	51.9	66.2	58.2	(-)	(-)	56.1	(-)	(-)	(-)	(-)	(-)	(-)
PN (Wang et al., 2019)	63.2	82.5	71.6	79.3	68.9	73.7	(-)	(-)	(-)	(-)	(-)	(-)
FASpell (Hong et al., 2019)	(-)	(-)	(-)	(-)	(-)	(-)	61.0	53.5	57.0	59.4	52.0	55.4
BERT	82.9	77.6	80.2	96.8	75.2	84.6	**65.6**	68.1	66.8	**63.1**	65.5	64.3
SpellGCN	**83.6**	**78.6**	**81.0**	**97.2**	**76.4**	**85.5**	65.1	**69.5**	**67.2**	**63.1**	**67.2**	**65.3**
SIGHAN 2015	D-P	D-R	D-F	C-P	C-R	C-F	D-P	D-R	D-F	C-P	C-R	C-F
LMC (Xie et al., 2015)	83.8	26.2	40.0	71.1	50.2	58.8	(-)	(-)	(-)	(-)	(-)	(-)
SL (Wang et al., 2018)	56.6	69.4	62.3	(-)	(-)	57.1	(-)	(-)	(-)	(-)	(-)	(-)
PN (Wang et al., 2019)	66.8	73.1	69.8	71.5	59.5	69.9	(-)	(-)	(-)	(-)	(-)	(-)
FASpell (Hong et al., 2019)	(-)	(-)	(-)	(-)	(-)	(-)	67.6	60.0	63.5	66.6	59.1	62.6
BERT	87.5	85.7	86.6	95.2	81.5	87.8	73.7	78.2	75.9	70.9	75.2	73.0
SpellGCN	**88.9**	**87.7**	**88.3**	**95.7**	**83.9**	**89.4**	**74.8**	**80.7**	**77.7**	**72.1**	**77.7**	**75.9**

Figure 5: The performance of SpellGCN. [4]

All in all, SpellGCN introduces external knowledge. We assume that errors only appear in the domain with phonological and visual similarity. With learning the embedding of these characters, we can find the nearest character to correct in the output stage by graph convolution. In this way, the corrected character will not have such randomness as the BERT model.

2.3 PLOME [5]

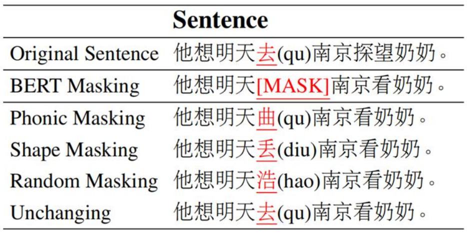
In this year, many papers on introducing phonological and visual similarity into Bert pre-training appeared in ACL. We picked up the paper published by Tencent AI Platform Department on ACL2021: ***PLOME: Pre-training with Misspelled Knowledge for Chinese Spelling Correction.*** (The reason we pick up this is they prove that the PLOME outperforms the above two method by comparative experiments.)

2.3.1 Motivation. The above paper [4] leveraged the confusion set, i.e., similar characters, to fuse the information of phonologically or visually similar characters. However, confusion is usually generated by heuristic rules or manual annotations; thus, its coverage is limited. The similarity was measured via rules rather than learned by the model. Therefore, such knowledge was not fully utilized.

2.3.2 Method. For the confusion set, the authors implemented it based on the masking strategy. The [MASK] strategy covers 15% of tokens. If the i-th token is chosen for the 15% [MASK] token, we replace it with the following strategy:

1. 60% : Random phonologically similar character.
2. 15% : Random visually similar character.
3. 15% : Unchanged i-th token.
4. 10% : Random token in the vocabulary.

The example of the [MASK] is shown in Figure 6.



	Sentence
Original Sentence	他想明天去(qu)南京探望奶奶。
BERT Masking	他想明天[MASK]南京看奶奶。
Phonic Masking	他想明天曲(qu)南京看奶奶。
Shape Masking	他想明天丢(diu)南京看奶奶。
Random Masking	他想明天浩(hao)南京看奶奶。
Unchanging	他想明天去(qu)南京看奶奶。

Figure 6: Examples of different masking strategies. [5]

For the embedding layer, this paper adopts character embedding, position embedding, phonetic embedding, and shape embedding. The character embedding and position embedding are consistent with the input of BERT. When constructing phonetic embedding, *Unihan Database* is used to obtain the character Pinyin mapping table (regardless of tone), and then Pinyin alphabet sequences of each character are input into the GRU network to obtain the Pinyin embedding vector. Similarly, when constructing shape embedding, use *Chaizi Database* to obtain the stroke order and then input the stroke order sequence into the GRU network to obtain the shape embedding vector of the character. An example is given in Figure 7.

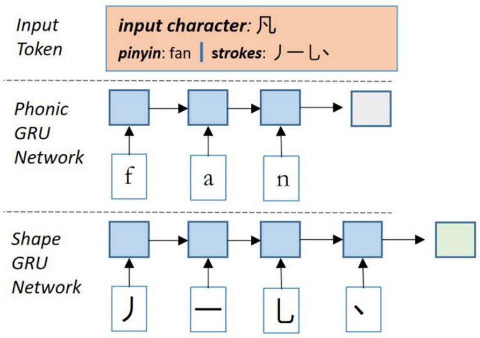


Figure 7: Illustration of phonic and shape GRU network. [5]

PLOME trains two tasks. One of them is character prediction, which predicts each replaced character in the input sentence like BERT. The other is pronunciation prediction: to learn the relevant knowledge of spelling error correction at the phonetic level, the paper takes spelling prediction as the pre-training task of PLOME, that is, to predict the correct pronunciation of the replaced character (there are about 430 different pronunciations in Chinese).

2.3.3 Result. The experiment was carried out on 'SIGHAN13', 'SIGHAN14', 'SIGHAN15', and 271K automatically generated samples. The experimental comparison was carried out in the two dimensions of character and sentence, respectively. Figure 9 shows the result table.

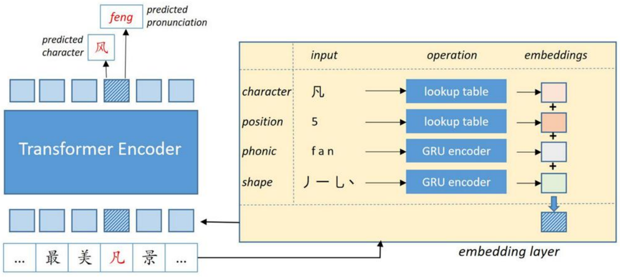
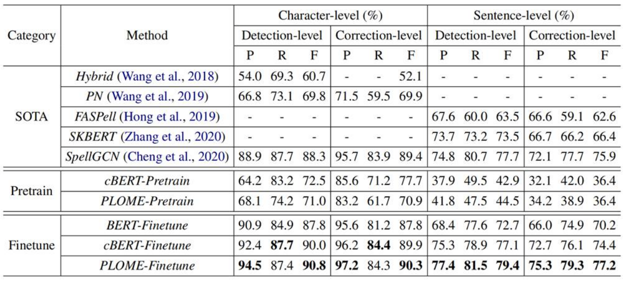


Figure 8: The framework of the proposed PLOME. [5]



Category	Method	Character-level (%)						Sentence-level (%)					
		Detection-level			Correction-level			Detection-level			Correction-level		
		P	R	F	P	R	F	P	R	F	P	R	F
SOTA	*Hybrid* (Wang et al., 2018)	54.0	69.3	60.7	-	-	52.1	-	-	-	-	-	-
	PN (Wang et al., 2019)	66.8	73.1	69.8	71.5	59.5	69.9	-	-	-	-	-	-
	FASPell (Hong et al., 2019)	-	-	-	-	-	-	67.6	60.0	63.5	66.6	59.1	62.6
	SKBERT (Zhang et al., 2020)	-	-	-	-	-	-	73.7	73.2	73.5	66.7	66.2	66.4
	SpellGCN (Cheng et al., 2020)	88.9	87.7	88.3	95.7	83.9	89.4	74.8	80.7	77.7	72.1	77.7	75.9
Pretrain	*cBERT-Pretrain*	64.2	83.2	72.5	85.6	71.2	77.7	37.9	49.5	42.9	32.1	42.0	36.4
	PLOME-Pretrain	68.1	74.2	71.0	83.2	61.7	70.9	41.8	47.5	44.5	34.2	38.9	36.4
Finetune	*BERT-Finetune*	90.9	84.9	87.8	95.6	81.2	87.8	68.4	77.6	72.7	66.0	74.9	70.2
	cBERT-Finetune	92.4	**87.7**	90.0	96.2	**84.4**	89.9	75.3	78.9	77.1	72.7	76.1	74.4
	PLOME-Finetune	**94.5**	87.4	**90.8**	**97.2**	84.3	**90.3**	**77.4**	**81.5**	**79.4**	**75.3**	**79.3**	**77.2**

Figure 9: The performance of PLOME. [5]

PLOME achieves better performance; however, the boost is still not such significant. In essence, these three models aim to correct the error model, and the way of introducing noise into knowledge is different. It is easy to understand why the three models do not improve significantly relative to BERT.

3 Another Readings

In addition to the three articles highlighted above, many other papers focus on CSC. (The reason they are not picked up is that the page number does not exceed 7.) In this section, we will briefly introduce the methods them partly.

3.1 V-style errors by the OCR [6]

This paper introduces OCR and ASR to construct shape-like and sound-like data, which has a significant effect on the detection model of LSTM. In terms of shape similarity, select a character in the correct sentence, add Gaussian noise to the graph of the character, and then use OCR to identify it. If it is different, it can be regarded as a wrong character pair.

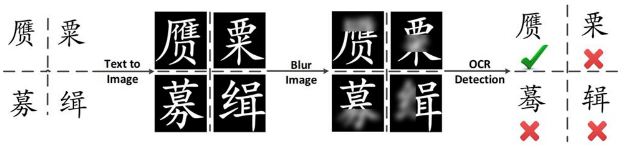


Figure 10: An example process of generating V-style errors by the OCR-based method. [6]

As shown in Figure 11, in the OCR detection results, except for 赝(yan4), the other three characters, i.e., 粟(li4), 募(mu4), and 缉(ji1), are incorrectly recognized as 栗(su4), 蓦(mo4), and 辑(ji2), respectively. All the three incorrect characters have similar shapes with their corresponding correct references. The experimental results show that the effect is remarkable and consistent with common sense; However, this task requires too much data. The gain of this complex construction method compared with the ordinary confusion set has not been determined, and the error correction application on OCR and ASR should be more suitable.

3.2 Confusionset-guided Pointer Networks [7]

This is a method based on the seq2seq model. Most of the input and output of errors are consistent, so if the seq2seq model is used, the copy mechanism is essential, and then the confusion set is introduced as a guide. When decoding, if the character on the thesaurus is predicted, select the confusion set of the original character at that position.

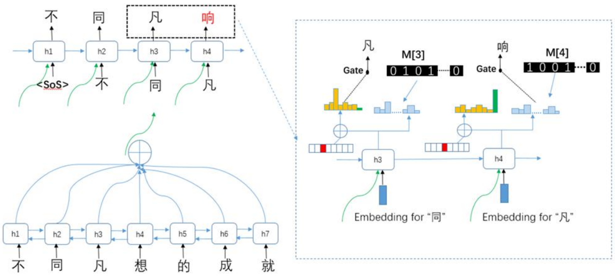


Figure 11: The framework of the proposed Confusionset-guided Pointer Networks. [7]

The experimental results show that the improvement is obvious after adding the confusion set. However, according to the recent research progress, the use of seq2seq in the field of CSC is a little short-sighted.

3.3 FASpell [8]

In the decoding stage, the original BERT method is to set different weights for multiple features. The FASpell method proposed in this paper uses both the confidence of context BERT and the similarity. Firstly, based on the training set, the scatter diagram of original-candidate similarity, and BERT confidence, the curve that can separate the true-detection-and-true-correction is drawn. All four plots in Figure 12 show the same confidence-similarity graph of candidates categorized by being true-detection-and-true-correction (T-d & T-c), true-detection-and-false-correction (T-d & F-c) and false-detection (F-d).

Using this method, the second candidate may also be selected. Intuitively, BERT's prediction is the most semantically possible character. The error words expressed by people are similar to the original words. With this information, the selection and sorting have coincided with the actual scene.

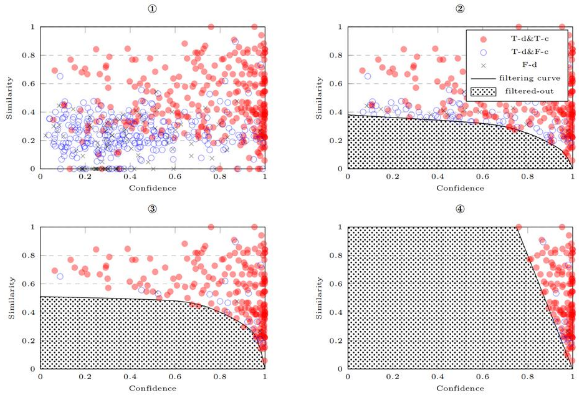


Figure 12: The confidence-similarity graph. [8]

4 Discussion

Chinese error correction, especially the task of character correction that does not involve grammar, is usually regarded as a problem that has been 'solved' in academia. That is, there are not many 'intelligent' components; as long as there are enough data, a good result can be obtained. However, the 'enough data' is a false proposition in the industry. Theoretically, there are countless kinds of error sentences. Human beings obtain the ability to judge by learning abstract knowledge such as grammar and logic. At the model level, how can we be regarded as a 'good' model? In my opinion, the problem of CSC is still a long way to proceed.

REFERENCES

[1] Liu, C.-L., et al. (2010). Visually and Phonologically Similar Characters in Incorrect Simplified Chinese Words, Beijing, China, Coling 2010 Organizing Committee.

[2] Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805.

[3] Zhang, S., Huang, H., Liu, J., & Li, H. (2020). Spelling error correction with soft-masked BERT. arXiv preprint arXiv:2005.07421.

[4] Cheng, X., Xu, W., Chen, K., Jiang, S., Wang, F., Wang, T., ... & Qi, Y. (2020). Spellgcn: Incorporating phonological and visual similarities into language models for chinese spelling check. arXiv preprint arXiv:2004.14166.

[5] Liu, S., Yang, T., Yue, T., Zhang, F., & Wang, D. (2021, August). PLOME: Pre-training with Misspelled Knowledge for Chinese Spelling Correction. In Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers) (pp. 2991-3000).

[6] Wang, D., Song, Y., Li, J., Han, J., & Zhang, H. (2018). A hybrid approach to automatic corpus generation for Chinese spelling check. In Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing (pp. 2517-2527).

[7] Wang, D., Tay, Y., & Zhong, L. (2019, July). Confusionset-guided pointer networks for Chinese spelling check. In Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics (pp. 5780-5785).

[8] Hong, Y., Yu, X., He, N., Liu, N., & Liu, J. (2019, November). Faspell: A fast, adaptable, simple, powerful chinese spell checker based on dae-decoder paradigm. In Proceedings of the 5th Workshop on Noisy User-generated Text (W-NUT 2019) (pp. 160-169).