BRIDGING THE GAP: A SELF-LEARNING MODEL USING IMPLICIT KNOWLEDGE FOR CHINESE SPELLING CORRECTION

Wenyao Cui, Jiahao Cai, Baohua Zhang, Yongyi Huang, Huaping Zhang[*]

School of Computer Science & Technology, Beijing Institute of Technology, China

ABSTRACT

Chinese Spelling Correction (CSC) is a challenging and essential task in natural language processing. In this study, we introduces a new method for Chinese Spelling Correction (CSC) that addresses three unattended areas in prior studies. Firstly, we use an Implicit Knowledge Extraction Network to overcome limitations of conventional methods that rely on explicit knowledge alone. Secondly, we use KL divergence to limit the effect of incorrect characters on semantic understanding, ensuring consistent meaning. Finally, we employ a Cor-Det framework rather than the traditional Det-Cor framework, offering more consistent learning objectives. Tests on three SIGHAN benchmarks show this method significantly surpassing baseline models, highlighting the crucial role of implicit knowledge in Chinese Spelling Correction tasks.

Index Terms— Chinese Spelling Correction, implicit knowledge

1. INTRODUCTION

Chinese spelling correction (CSC) is an important and difficult task that aims to detect and correct spelling errors in Chinese text. The CSC task requires models with strong language comprehension and reasoning ability, which makes it one of the most difficult tasks in natural language processing.

Previous works [1, 2, 3] did not adequately consider the impact of erroneous characters on the correct semantic understanding of sentences. This oversight may lead to difficulty in distinguishing and rectifying errors, which could, in turn, affect the reliability and efficiency of the CSC model.

Previous works [1, 4] generally used a Det-Cor framework, in which the detection of error characters is performed first, and then correction is performed based on the detection results. However, humans first correct errors and then obtain the positions of the error characters based on the comparison of the corrected results with the original text. Under the Det-Cor framework, the misdetected characters can hardly be corrected.

Previous approaches always incorporate explicit knowledge into the correction. [5, 6, 7] used rules as knowledge for correction. [1, 2] achieved good results by using learned knowledge from large pre-trained language models (PLMs) such as BERT [8]. [9, 3, 4, 10] incorporated external phonetic and graphic information of characters as knowledge.

Wrong:	郑州是一个急束(shu)发展的城市，它的GDP很低。
Baseline:	郑州是一个急速(su)发展的城市，它的GDP很低。
Correct:	郑州是一个急需(xu)发展的城市，它的GDP很低。
Translation:	Zhengzhou is a city in desperate need of development, and it has a very low GDP.
Wrong:	我听说这个礼拜六你要开一个误(wu)会。
Baseline:	我听说这个礼拜六你要开一个聚(ju)会。
Correct:	我听说这个礼拜六你要开一个舞(wu)会。
Translation:	I heard you're having a dance this Saturday.

Table 1. Examples of Chinese spelling correction. The incorrect and correct characters are marked in red and blue, respectively.

Explicit knowledge leads the model to rely more on it to make a decision. For example, PLMs based models will predict most common characters, and models that incorporate external phonetic and graphic knowledge tend to make error corrections that share the similar pronunciation or shape.

As shown in the upper example in Table 1, the pronunciation and shape of the character “束” (bunch, pronounced "shu") makes the model miscorrect the character to “速” (speed, pronounced "su"), because of their similar pronunciation. However, we should correct it to “需” (need, pronounced "xu") according to the context. This miscorrection is the result of explicit knowledge of the phonetic and graphic information of the characters. In the lower example in Table 1, PLMs based models tend to miscorrect the character “误” (mistake, pronounced "wu") to “聚” (gather, pronounced "ju") instead of “舞” (dance, pronounced "wu"). This is because “聚会” (party) is more common than “舞会” (dance party) for PLMs. Therefore, effectively extracting and utilizing the implicit knowledge for correction is the key to enhancing the Chinese Spelling Correction task.

[*]This is the Corresponding Author, Email: kevinzhang@bit.edu.cn

The contributions of this paper are as follows:

1) **Bridging the Gap for Correction:** By introducing a Implicit Knowledge Extraction Network, we efficiently extract valuable implicit knowledge for Correction. This approach effectively compensates for the limitations of conventional models that focus primarily on explicit knowledge, resulting in broader coverage of possible error scenarios and reducing the risk of miscorrections.

2) **Constraining Semantic Consistency:** We develop a unique approach utilizing KL divergence to minimize the impact of erroneous characters on the semantic understtanding of CSC tasks. This contribution advances the understanding and effectiveness of error-prone character handling compared to previous methods.

3): **Innovative Correction-Detection framework:** Our study presents a novel Correction-Detection (Cor-Det) framework, an improvement upon the conventional Detection-Correction (Det-Cor) framework. This new framework establishes consistent learning objectives for error detection and correction, leading to more efficient and optimized solutions for CSC tasks.

4) **Our method achieves state-of-the-art results:** Our method achieves state-of-the-art results on the SIGHAN benchmarks, which also illustrates the importance of implicit knowledge in correction tasks.

2. RELATED WORKS

With the great success of PLMs (e.g., BERT [8]), it is intuitive to use PLMs to capture explicit phonetic and graphic knowledge of Chinese characters. SpellGCN [2] incorporated phonetic and graphic knowledge into BERT through a specialized graph convolutional network. PLOME [9], which was proposed to be a task-specific pre-trained language model for CSC, used a confusion set based masking strategy and introduced various external knowledge. In addition, some studies [10, 3] utilized phonetic and graphic knowledge to model the similarities of the characters for correction. Phonetic MLM [4] pre-trained a masked language model with phonetic features to improve the model's ability to understand sentences with misspellings. [11] promoted the improvement of the CSC task by narrowing the gap between the knowledge of PLMs and the goal of CSC.

Some previous studies used an error detector as the preliminary step for correction, which turns the CSC into a multi-task problem. Soft-Masked BERT [1] leveraged a cascading architecture in which BiGRU was used to detect error positions and BERT was used to predict correct characters. [12] proposed a two-stage cloze-style detector-corrector framework for correction. [13] predicted characters via the fusion of hidden states from a correction module and a detection module.

3. OUR APPROACH

3.1. Problem Definition

The CSC task can be represented by the following: given a sequence $X = (x_1, x_2, ..., x_n)$, our goal is to generate a sequence $Y = (y_1, y_2, ..., y_n)$ of the same length, where the incorrect characters in X will be replaced by correct characters to form the correct sentence Y.

3.2. The SLIK Model

We propose a novel neural network model called SLIK for CSC, as illustrated in Figure 1. SLIK is composed of 1) an Implicit Knowledge Extraction Network, 2) a Correction Network based on BERT, and 3) a Detection Network based on Multi-Head Attention.

3.3. Implicit Knowledge Extraction Network

In this paper, we use FFN as our implementation. First, we input the error sentence embedding $E_e = (e_1, e_2, ..., e_n)$ into the Implicit Knowledge Extraction Network, where e_i denotes the embedding of character x_i, which is the sum of word embedding, position embedding, and segment embedding of the character, as in BERT. We use K to denote the extracted knowledge.

$$K = KE(E_e) \tag{1}$$

3.4. Correction Network

The inputs of the Correction Network are knowledge K extracted by the implicit knowledge extraction network and the embeddings of the error sentences $E_e = (e_1, e_2, ..., e_n)$. The outputs are the probability distributions of the character vocabulary.

As illustrated in Figure 1, the Correction Network has two paths: the positive path that incorporates K is represented by Pos, and the negative path that does not incorporate K is represented by Neg. We use E_e plus K to denote incorporating knowledge and E_e minus K to denote not incorporating knowledge. That is:

$$\begin{aligned} E_{Pos} &= E_e + K \\ E_{Neg} &= E_e - K \end{aligned} \tag{2}$$

where E_{Pos} denotes the sentence representation that incorporates knowledge and E_{Neg} denotes the sentence representation that does not. We then pass E_{Pos} and E_{Neg} through the Encoder module of BERT to get the hidden states H_{Pos} and H_{Neg}, respectively:

$$\begin{aligned} H_{Pos} &= Encoder(E_{Pos}) + K \\ H_{Neg} &= Encoder(E_{Neg}) - K \end{aligned} \tag{3}$$

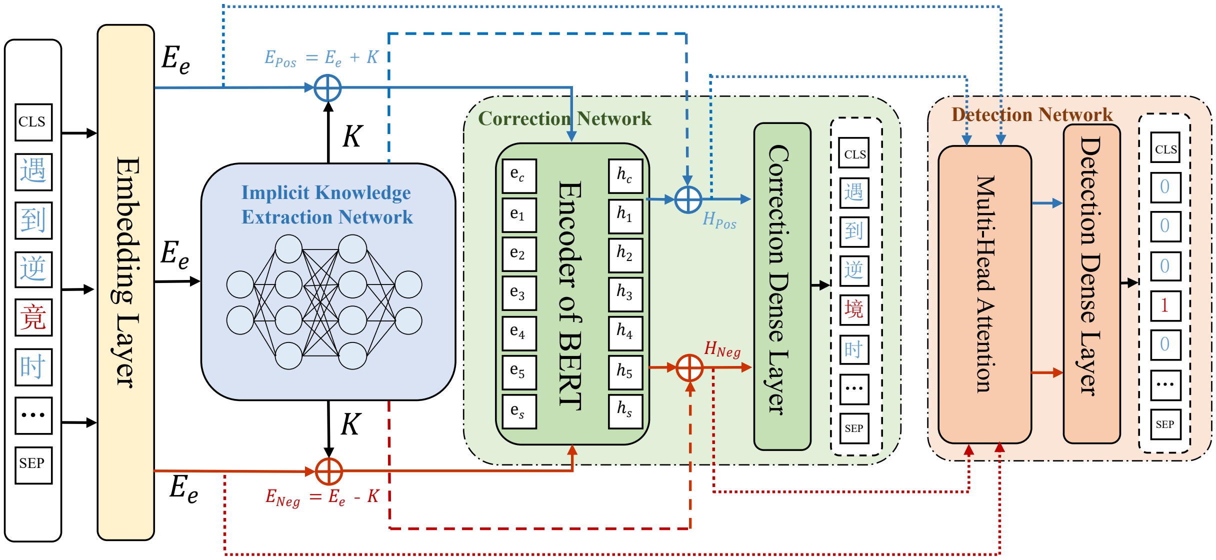

Fig. 1. SLIK has 3 networks: Knowledge Extraction, Correction, and Detection. Implicit knowledge is extracted and passed to Correction, then to Detection. The process has upper (Pos) and lower (Neg) paths, where Pos incorporates knowledge.

Then we map the vectors into the character vocabulary space by a fully connected layer and exploit a Softmax function to obtain the probability distributions $\tilde{Y}^c_{pos}$ and $\tilde{Y}^c_{neg}$ of the characters over the character vocabulary at each position in the sentence:

$$\begin{aligned} \tilde{Y}^c_{pos} &= Softmax(FC(H_{Pos})) \\ \tilde{Y}^c_{neg} &= Softmax(FC(H_{Neg})) \end{aligned} \tag{4}$$

3.5. Detection Network

Detection Network base on Multi-Head Attention, the Query is hidden state vectors H_{Pos} for Pos and H_{Neg} for Neg, the Key and the Value is error sentence embedding E_e

The Detection Network is a sequential binary labeling module. The inputs of the Detection Network are error sentence embeddings $E_e = (e_1, e_2, ..., e_n)$ with hidden states H_{Pos}, as well as with H_{Neg}, respectively. The output is the probability that a position is incorrect.

After obtaining the hidden state vectors H_{Pos} and H_{Neg} of Pos and Neg, we compare these two vectors with the error sentence vector E_e respectively to obtain the detection results of Pos and Neg, which can be represented as $\tilde{Y}^d_{Pos} = (\tilde{y}^d_{Pos,1}, \tilde{y}^d_{Pos,2}, ..., \tilde{y}^d_{Pos,n})$ and $\tilde{Y}^d_{Neg} = (\tilde{y}^d_{Neg,1}, \tilde{y}^d_{Neg,2}, ...\tilde{y}^d_{Neg,n})$:

$$\begin{aligned} \tilde{Y}^d_{Pos} = \sigma(FC(MHA(&Query = H_{Pos}, \\ &Key = E_e, Value = E_e)) \\ \tilde{Y}^d_{Neg} = \sigma(FC(MHA(&Query = H_{Neg}, \\ &Key = E_e, Value = E_e)) \end{aligned} \tag{5}$$

where σ represents the sigmoid function.

3.6. Learning Objectives

Correction objective For Pos, we expect it to make a correct correction, and for Neg, we expect it not to make a correct correction. $Correct = (x^{correct}_1, ..., x^{correct}_n)$ represents the correct sentence, and $Error = (x^{error}_1, ..., x^{error}_n)$ represents the error sentence. That is:

$$\begin{aligned} \mathcal{L}^c_{Pos} &= NLLLoss(\tilde{Y}^c_{pos}, Correct) \\ \mathcal{L}^c_{Neg} &= NLLLoss(\tilde{Y}^c_{neg}, Error) \end{aligned} \tag{6}$$

Detection objective

For Pos, the target value of error detection probability is 1 on the positions of spelling errors and 0 on the rest, whereas for Neg, the target probability is 0 on all positions in a sentence since we do not want it to make corrections. We use det^{label}_{Pos} to represent the detection target for Pos and det^{label}_{Neg} to represent the detection target for Neg. Then we have:

$$\begin{aligned} \mathcal{L}^d_{Pos} &= BCELoss(\tilde{Y}^d_{pos}, det^{label}_{Pos}) \\ \mathcal{L}^d_{Neg} &= BCELoss(\tilde{Y}^d_{Neg}, det^{label}_{Neg}) \end{aligned} \tag{7}$$

Semantic consistency Given that the difference of the targets between Pos and Neg is limited to the error characters' positions in a sentence, the probability distributions of Pos and Neg for the correct characters' positions in a sentence should be as consistent as possible.

$$\mathcal{L}_{KL} = \sum_{i}^{n} \tilde{y}^{d}_{Pos,i}(log(\frac{\tilde{y}^{d}_{Pos,i}}{\tilde{y}^{d}_{Neg,i}})) + \sum_{i}^{n} \tilde{y}^{d}_{Neg,i}(log(\frac{\tilde{y}^{d}_{Neg,i}}{\tilde{y}^{d}_{Pos,i}})) \quad (8)$$

where n represents the length of a sentence and i represents positions of correct characters in a sentence.

3.7. Joint Learning

We simply add the above losses together and find that this gives good results:

$$\mathcal{L} = \mathcal{L}^{c}_{Pos} + \mathcal{L}^{c}_{Neg} + \mathcal{L}^{d}_{Pos} + \mathcal{L}^{d}_{Neg} + \mathcal{L}_{KL} \quad (9)$$

4. EXPERIMENTS

4.1. Datasets and Evaluation Metrics

To compare SLIK with state-of-the-art methods, we conduct tests on three SIGHAN benchmarks: SIGHAN13/14/15. We include the Wang271K dataset [14] for training in addition to the training set of SIGHAN itself. The sentence-level precision, recall, and F1 score are reported as the evaluation metrics, as in most of the previous works.

Dataset	Method	Detection Level				Correction Level			
		Acc	Pre	Rec	F1	Acc	Pre	Rec	F1
SIGHAN13	REALISE [3]	82.7	88.6	82.5	85.4	81.4	87.2	81.2	84.1
	MDCSpell [13]	-	89.1	78.3	83.4	-	87.5	76.8	81.8
	SLIK(ours)	**84.5**	**89.6**	**84.2**	**86.8**	**83.7**	**88.7**	**83.4**	**85.9**
SIGHAN14	REALISE [3]	78.4	67.8	71.5	69.6	77.7	66.3	70.0	68.1
	MDCSpell [13]	-	70.2	68.8	69.5	-	69.0	67.7	68.3
	SLIK(ours)	**82.8**	**71.4**	**72.4**	**71.9**	**82.2**	**70.0**	70.9	**70.4**
SIGHAN15	REALISE [3]	84.7	77.3	81.3	79.3	84.0	75.9	79.9	77.8
	MDCSpell [13]	-	80.8	80.6	80.7	-	78.4	78.2	78.3
	SLIK(ours)	**87.9**	**81.4**	82.5	**82.0**	**86.9**	**79.2**	80.4	**79.8**

Table 2. The performance of our model and all baseline models on SIGHAN benchmarks. The results show that SLIK outperforms baseline models on the three SIGHAN benchmarks.

4.2. Main Results

Table 2 presents the experimental results of all methods on three SIGHAN benchmarks. All the results show that although external phonetic and graphic knowledge is beneficial to the CSC task, our proposed semantic consistency constraint, implicit knowledge extraction method and Cor-Det framework are more beneficial to the CSC task. Semantic consistency constraint between correct and incorrect sentences minimizes the impact of error-prone characters on semantic understanding. The Cor-Det framework improves learning objective consistency for error detection and correction. Implicit knowledge compensates for the gap between the learned knowledge of PLMs and the knowledge required for the CSC task.

Dataset	Method	Correction Level F1
SIGHAN13	SLIK	85.9
	-Det	84.3
	-K	84.0
	-Semantic Consistency	83.7
SIGHAN14	SLIK	70.4
	-Det	69.9
	-K	65.6
	-Semantic Consistency	69.4
SIGHAN15	SLIK	79.8
	-Det	79.6
	-K	75.1
	-Semantic Consistency	78.4

Table 3. Ablation results of SLIK model on SIGHAN benchmarks focusing on Correction Level F1 score.

4.3. Ablation Study

Table 3 shows the ablation study results on three aspects: 1) using implicit knowledge, 2) mitigating erroneous characters' impact, and 3) implementing the Correction-Detection framework. Disabling semantic consistency leads to a decline in F1 scores, affirming its importance in Chinese Spelling Correction tasks. The model's F1 score drops by 4% without the Implicit Knowledge Extraction Network, underscoring its cruciality. Any removed Correction-Detection framework-related component also decreases performance, highlighting each component's efficiency. The findings verify the necessity of these three aspects and support the SLIK model's effectiveness in advancing Chinese Spelling Correction.

5. CONCLUSION

In this paper, we propose a novel end-to-end method called SLIK for Chinese Spelling Correction (CSC), concentrating on three main contributions: 1. Extracting implicit knowledge. Traditional methods relied on explicit knowledge, such as external knowledge which do not cover all situations and may lead to miscorrections. SLIK focuses on extracting crucial implicit knowledge to ensure more effective error correction. 2. Addressing the impact of erroneous characters on semantic understanding in CSC tasks by constraining semantic consistency. Our method mitigates the effects of incorrect characters on the accurate understanding of semantics, overcoming limitations in prior approaches that failed to consider this crucial aspect. 3. Implementing the Correction-Detection (Cor-Det) framework, enabling consistent learning objectives for both correction and detection within SLIK. Our experiments on three SIGHAN benchmarks demonstrate the effectiveness of addressing these key contributions by showcasing that the SLIK model, incorporating implicit knowledge, significantly outperforms baseline models relying on explicit knowledge.

6. REFERENCES

[1] Shaohua Zhang, Haoran Huang, Jicong Liu, and Hang Li, “Spelling error correction with soft-masked BERT,” in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, Online, July 2020, pp. 882–890, Association for Computational Linguistics.

[2] Xingyi Cheng, Weidi Xu, Kunlong Chen, Shaohua Jiang, Feng Wang, Taifeng Wang, Wei Chu, and Yuan Qi, “Spellgcn: Incorporating phonological and visual similarities into language models for chinese spelling check,” *arXiv preprint arXiv:2004.14166*, 2020.

[3] H. D. Xu, Z. Li, Q. Zhou, C. Li, and X. L. Mao, “Read, listen, and see: Leveraging multimodal information helps chinese spell checking,” *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, 2021.

[4] Ruiqing Zhang, Chao Pang, Chuanqiang Zhang, Shuohuan Wang, Zhongjun He, Yu Sun, Hua Wu, and Haifeng Wang, “Correcting chinese spelling errors with phonetic pre-training,” in *Findings of the Association for Computational Linguistics: ACL/IJCNLP 2021, Online Event, August 1-6, 2021*, Chengqing Zong, Fei Xia, Wenjie Li, and Roberto Navigli, Eds. 2021, vol. ACL/IJCNLP 2021 of *Findings of ACL*, pp. 2250–2261, Association for Computational Linguistics.

[5] Junjie Yu and Zhenghua Li, “Chinese spelling error detection and correction based on language model, pronunciation, and shape,” in *Proceedings of the Third CIPS-SIGHAN Joint Conference on Chinese Language Processing*, Wuhan, China, Oct. 2014, pp. 220–223, Association for Computational Linguistics.

[6] Wei-Cheng Chu and Chuan-Jie Lin, “NTOU Chinese spelling check system in sighan-8 bake-off,” in *Proceedings of the Eighth SIGHAN Workshop on Chinese Language Processing*, Beijing, China, July 2015, pp. 137–143, Association for Computational Linguistics.

[7] Tao-Hsing Chang, Hsueh-Chih Chen, and Cheng-Han Yang, “Introduction to a proofreading tool for Chinese spelling check task of SIGHAN-8,” in *Proceedings of the Eighth SIGHAN Workshop on Chinese Language Processing*, Beijing, China, July 2015, pp. 50–55, Association for Computational Linguistics.

[8] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova, “BERT: Pre-training of deep bidirectional transformers for language understanding,” in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, Minneapolis, Minnesota, June 2019, pp. 4171–4186, Association for Computational Linguistics.

[9] Shulin Liu, Tao Yang, Tianchi Yue, Feng Zhang, and Di Wang, “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)*, 2021, pp. 2991–3000.

[10] Li Huang, Junjie Li, Weiwei Jiang, Zhiyu Zhang, Minchuan Chen, Shaojun Wang, and Jing Xiao, “PHMOSpell: Phonological and morphological knowledge guided Chinese spelling check,” 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)*, Online, Aug. 2021, pp. 5958–5967, Association for Computational Linguistics.

[11] Yinghui Li, Qingyu Zhou, Yangning Li, Zhongli Li, Ruiyang Liu, Rongyi Sun, Zizhen Wang, Chao Li, Yunbo Cao, and Hai-Tao Zheng, “The past mistake is the future wisdom: Error-driven contrastive probability optimization for chinese spell checking,” *arXiv preprint arXiv:2203.00991*, 2022.

[12] Jing Li, Dafei Yin, Haozhao Wang, and Yonggang Wang, “Dcspell: A detector-corrector framework for chinese spelling error correction,” *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2021.

[13] Chenxi Zhu, Ziqiang Ying, Boyu Zhang, and Feng Mao, “MDCSpell: A multi-task detector-corrector framework for Chinese spelling correction,” in *Findings of the Association for Computational Linguistics: ACL 2022*, Dublin, Ireland, May 2022, pp. 1244–1253, Association for Computational Linguistics.

[14] Dingmin Wang, Yan Song, Jing Li, Jialong Han, and Haisong Zhang, “A hybrid approach to automatic corpus generation for Chinese spelling check,” in *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, Brussels, Belgium, Oct.-Nov. 2018, pp. 2517–2527, Association for Computational Linguistics.