Read, Listen, and See: Leveraging Multimodal Information Helps Chinese Spell Checking

Heng-Da Xu[1]*, **Zhongli Li**[2]*, **Qingyu Zhou**[2], **Chao Li**[2], **Zizhen Wang**[2], **Yunbo Cao**[2], **Heyan Huang**[1], **Xian-Ling Mao**[1]
[1]Beijing Institute of Technology
[2]Tencent Cloud Xiaowei
{xuhengda,hhy63,maoxl}@bit.edu.cn
{neutrali,qingyuzhou,diegoli,zizhenwang,yunbocao}@tencent.com

Abstract

Chinese Spell Checking (CSC) aims to detect and correct erroneous characters for user-generated text in Chinese language. Most of the Chinese spelling errors are misused semantically, phonetically or graphically similar characters. Previous attempts notice this phenomenon and try to utilize the similarity relationship for this task. However, these methods use either heuristics or handcrafted confusion sets to predict the correct character. In this paper, we propose a Chinese spell checker called REALISE, by directly leveraging the multimodal information of the Chinese characters. The REALISE model tackles the CSC task by (1) capturing the semantic, phonetic and graphic information of the input characters, and (2) selectively mixing the information in these modalities to predict the correct output. Experiments[1] on the SIGHAN benchmarks show that the proposed model outperforms strong baselines by a large margin.

1 Introduction

The Chinese Spell Checking (CSC) task aims to identify erroneous characters and generate candidates for correction. It has attracted much research attention, due to its fundamental and wide applications such as search query correction (Martins and Silva, 2004; Gao et al., 2010), optical character recognition (OCR) (Afli et al., 2016), automatic essay scoring (Dong and Zhang, 2016). Recently, rapid progress (Zhang et al., 2020; Cheng et al., 2020) has been made on this task, because of the success of large pretrained language models (Devlin et al., 2019; Liu et al., 2019; Yang et al., 2019).

In alphabetic languages such as English, spelling errors often occur owing to one or more wrong characters, resulting in the written word not in the dictionary problem (Tachibana and Komachi, 2016). However, Chinese characters are valid if they can be typed in computer systems, which causes that the spelling errors are de facto misused characters in the context of computer-based language processing. Considering the formation of Chinese characters, a few of them were originally pictograms or phono-semantic compound characters (Jerry, 1988). Thus, in Chinese, the spelling errors are not only the misused characters with confusing semantic meaning, but also the characters which are phonetically or graphically similar (Liu et al., 2010, 2011). Table 1 shows two examples of Chinese spelling error. In the first example, phonetic information of "平" (flat) is needed to get the correct character "瓶" (bottle) since they share the same pronunciation "píng". The second example needs not only phonetic, but also graphic information of the erroneous character "轻" (light). The

Phonetically Similar Case		
Sent.	晚饭后他递给我一平(píng, *flat*)红酒。	
Cand.	晚饭后他递给我一杯(bēi, *cup*)红酒。	✗
	晚饭后他递给我一瓶(píng, *bottle*)红酒。	✓
	晚饭后他递给我一箱(xiāng, *box*)红酒。	✗
Trans.	He handed me a *bottle* of red wine after dinner.	
Graphically Similar Case		
Sent.	每天放学我都会轻(qīng, *light*)过这片树林。	
Cand.	每天放学我都会路(lù, *pass*)过这片树林。	✗
	每天放学我都会经(jīng, *go*)过这片树林。	✓
	每天放学我都会走(zǒu, *walk*)过这片树林。	✗
Trans.	I *go* through this wood every day after school.	

Table 1: Two examples of Chinese spelling errors and their candidate corrections. "Sent./Cand./Trans." are short for sentence/candidates/translation respectively. The wrong/candidate/correct characters with their pronunciation and translation are in red/orange/blue color.

*Heng-Da Xu and Zhongli Li contributed equally. Work is done during internship at Tencent Cloud Xiaowei. Qingyu Zhou is the corresponding author.

[1]Code and model are available at https://github.com/DaDaMrX/ReaLiSe.

correct one, "经" (go), has the same right radical as "轻" and similar pronunciation ("qīng" and "jīng"). Therefore, considering the intrinsic nature of Chinese, it is essential to leverage the phonetic and graphic knowledge of the Chinese characters along with the textual semantics for the CSC task.

In this paper, we propose REALISE (**Rea**d, **Li**sten, and **S**ee), a Chinese spell checker which leverages the semantic, phonetic and graphic information to correct the spelling errors. The REALISE model employs three encoders to learn informative representations from textual, acoustic and visual modalities. First, BERT (Devlin et al., 2019) is adopted as the backbone of the semantic encoder to capture the textual information. For the acoustic modality, *Hanyu Pinyin* (pinyin), the romanization spelling system for the sounds of Chinese characters, is used as the phonetic features. We design a hierarchical encoder to process the pinyin letters at the character-level and the sentence-level. Meanwhile, for the visual modality, we build character images with multiple channels as the graphic features, where each channel corresponds to a specific Chinese font. Then, we use ResNet (He et al., 2016) blocks to encode the images to get the graphic representation of characters.

With the representation of three different modalities, one challenge is how to fuse them into one compact multimodal representation. To this end, a selective modality fusion mechanism is designed to control how much information of each modality can flow to the mixed representation. Furthermore, as the pretrain-finetune procedure has been proven to be effective on various NLP tasks (Devlin et al., 2019; Dong et al., 2019; Sun et al., 2020), we propose to pretrain the phonetic and the graphic encoders by predicting the correct character given input in the corresponding modality.

We conduct experiments on the SIGHAN benchmarks (Wu et al., 2013; Yu et al., 2014; Tseng et al., 2015). By leveraging multimodal information, REALISE outperforms all previous state-of-the-art models by a large margin. Compared to previous methods using confusion set (Lee et al., 2019) to capture the character similarity relationships, such as the SOTA SpellGCN (Cheng et al., 2020), REALISE achieves an averaging 2.4% and 2.6% F1 improvements at detection-level and correction-level. Further analysis shows that our model performs better on the errors which are not defined in the handcrafted confusion sets. This indicates that leveraging the phonetic and graphic information of Chinese characters can better capture the easily-misused characters.

To summarize, the contributions of this paper include: (i) we propose to leverage phonetic and graphic information of Chinese characters besides textual semantics for the CSC task; (ii) we introduce the selective fusion mechanism to integrate multimodal information; (iii) we propose acoustic and visual pretraining tasks to further boost the model performance; (iv) to the best of our knowledge, the proposed REALISE model achieves the best results on the SIGHAN CSC benchmarks.

2 Related Work

2.1 Chinese Spell Checking

The CSC task is to detect and correct spelling errors in Chinese sentences. Early works design various rules to deal with different errors (Chang et al., 2015; Chu and Lin, 2015). Next, traditional machine learning algorithms are brought to this field, such as Conditional Random Field and Hidden Markov Model (Wang and Liao, 2015; Zhang et al., 2015). Then, neural-based methods have made great progress in CSC. Wang et al. (2018) treat the CSC task as a sequence labeling problem, and use a bidirectional LSTM to predict the correct characters. With the great success of large pretrained language models (e.g., BERT (Devlin et al., 2019)), Hong et al. (2019) propose the FASpell model, which use a BERT-based denoising autoencoder to generate candidate characters and uses some empirical measures to select the most likely ones. Besides, the Soft-Masked BERT model (Zhang et al., 2020) leverages a cascading architecture where GRU is used to detect the erroneous positions and BERT is used to predict correct characters.

Previous works (Yu and Li, 2014; Wang et al., 2019; Cheng et al., 2020) using handcrafted Chinese character confusion set (Lee et al., 2019) aim to correct the errors by discovering the similarity of the easily-misused characters. Wang et al. (2019) leverage the pointer network (Vinyals et al., 2015) by picking the correct character from the confusion set. Cheng et al. (2020) propose a SpellGCN model which models the character similarity through Graph Convolution Network (GCNs) (Kipf and Welling, 2016) on the confusion set. However, the character confusion set is predefined and fixed, which cannot cover all the similarity relations, nor can it distinguish the divergence in the similarity

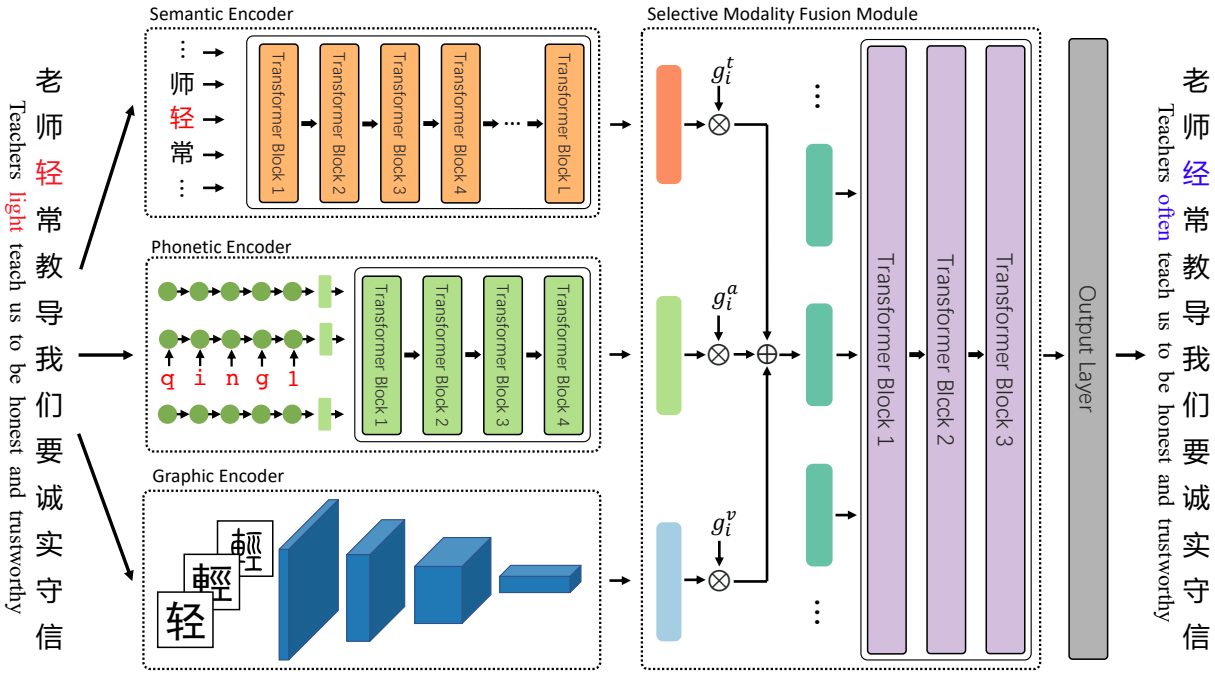

Figure 1: Architecture overview of the REALISE model. The semantic, phonetic and graphic encoders, are used to capture the information in textual, acoustic and visual modalities. The fusion module selectively fuses the information from three encoders. In the example input, to correct the erroneous character, "轻" (qīng, light), we need not only the contextual text information, but also the phonetic and graphic information of the character itself.

of Chinese characters. In this work, we discard the predefined confusion set and directly use the multimodal information to discover the subtle similarity relationship between all Chinese characters.

2.2 Multimodal Learning

There has been much research to integrate information from different modalities to achieve better performance. Tasks such as Multimodal Sentiment Analysis (Zadeh et al., 2016; Zhang et al., 2019), Visual Question Answering (Antol et al., 2015; Chao et al., 2018) and Multimodal Machine Translation (Hitschler et al., 2016; Barrault et al., 2018) have made much progress. Recently, multimodal pretraining models have been proposed, such as VL-BERT (Su et al., 2020), Unicoder-VL (Li et al., 2020), and LXMERT (Tan and Bansal, 2019). In order to incorporate the visual information of Chinese characters into language models, Meng et al. (2019) design a Tianzige-CNN to facilitate some NLP tasks, such as named entity recognition and sentence classification. To the best of our knowledge, this paper is the first work to leverage multimodal information to tackle the CSC task.

3 The REALISE Model

In this section, we introduce the REALISE model, which utilizes the semantic, phonetic, and graphic information to distinguish the similarities of Chinese characters and correct the spelling errors. As shown in Figure 1, multiple encoders are firstly employed to capture valuable information from textual, acoustic and visual modalities. Then, we develop a selective modality fusion module to obtain the context-aware multimodal representations. Finally, the output layer predicts the probabilities of error corrections.

3.1 The Semantic Encoder

We adopt BERT (Devlin et al., 2019) as the backbone of the semantic encoder. BERT provides rich contextual word representation with the unsupervised pretraining on large corpora.

The input tokens $\mathbf{X} = (x_1, \ldots, x_N)$ are first projected into $\mathbf{H}_0^t$ through the input embedding. Then the computation of Transformer (Vaswani et al., 2017) encoder layers can be formulated as:

$$\mathbf{H}_l^t = \text{Transformer}_l(\mathbf{H}_{l-1}^t), l \in [1, L] \quad (1)$$

where L is the number of Transformer layers. Each layer consists of a multi-head attention module

and a feed-forward network with the residual connection (He et al., 2016) and layer normalization (Ba et al., 2016). The output of the last layer $\mathbf{H}^t = \mathbf{H}^t_L = (h^t_1, \ldots, h^t_N)$ is used as the contextualized semantic representation of the input tokens in textual modality.

3.2 The Phonetic Encoder

Hanyu Pinyin (pinyin) is the romanization for Chinese to "spell out" the sounds of characters. We use it to calculate the phonetic representation in this paper. The pinyin of a Chinese character consists of three parts: initial, final, and tone. The initial (21 in total) and final (39 in total) are written with letters in the English alphabet. The 5 kinds of tones (take the final "a" as an example, { ā, á, ǎ, à, a }) can be mapped into numbers $\{1, 2, 3, 4, 0\}$. Though the vocabulary size of pinyin for all Chinese characters is a fixed number, we use a sequence of letters in REALISE to capture the subtle phonetic difference between Chinese characters. For example, the pinyin of "中" (middle) and "棕" (brown) are "zhōng" and "zōng" respectively. The two characters have very similar sounds but quite different meanings. We thus represent pinyin as a symbol sequence, e.g., $\{z, h, o, n, g, 1\}$ for "中". We denote the pinyin of the i-th character in the input sentence as $\boldsymbol{p}_i = (p_{i,1}, \ldots, p_{i,|\boldsymbol{p}_i|})$, where $|\boldsymbol{p}_i|$ is the length of pinyin $\boldsymbol{p}_i$.

In REALISE, we design a hierarchical phonetic encoder, which consists of a character-level encoder and a sentence-level encoder.

The Character-level Encoder is to model the basic pronunciation and capture the subtle sound difference between characters. It is a single-layer uni-directional GRU (Cho et al., 2014), which encodes the pinyin of the i-th character x_i as:

$$\tilde{h}^a_{i,j} = \text{GRU}(\tilde{h}^a_{i,j-1}, E(p_{i,j})) \tag{2}$$

where $E(p_{i,j})$ is the embedding of the pinyin symbol $p_{i,j}$, and $\tilde{h}^a_{i,j}$ is the j-th hidden states of the GRU. The last hidden state is used as the character-level phonetic representation of x_i.

The Sentence-level Encoder is a 4-layer Transformer with the same hidden size as the semantic encoder. It is designed to obtain the contextualized phonetic representation for each Chinese character. As the independent phonetic vectors are not distinguished in order, we add the positional embedding to each vector in advance. Then, we pack these phonetic vectors together, and apply the Transformer layers to calculate the contextualized representation in acoustic modality, denoted as $\mathbf{H}^a = (h^a_1, h^a_2, ..., h^a_N)$. Note that owing to the Transformer architecture, this representation is also normalized.

3.3 The Graphic Encoder

We apply the ResNet (He et al., 2016) as the graphic encoder. The graphic encoder has 5 layers of ResNet blocks (denoted as ResNet5) followed by a layer normalization (Ba et al., 2016) operation. We formulate this procedure as follows:

$$\begin{aligned} \tilde{h}^v_i &= \text{ResNet5}(\mathbf{I}_i) \\ h^v_i &= \text{LayerNorm}(\tilde{h}^v_i) \end{aligned} \tag{3}$$

where $\mathbf{I}_i$ is the image of the i-th character x_i in the input sentence, and LayerNorm means layer normalization.

In order to extract graphic information effectively, each block in ResNet5 halves the width and height of the image, and increases the number of channels. Thus, the final output is a vector with the length equal to the number of output channels, i.e., both height and width become 1. Furthermore, we set the number of output channels to the hidden size in the semantic encoder for the follow-up modality fusion. We denote the representation in visual modality of the input sentence as $\mathbf{H}^v = (h^v_1, h^v_2, \ldots, h^v_N)$.

The character image of x_i is read from preset font files. Since the scripts of Chinese characters have evolved for thousands of years, to capture the graphic relationship between character as much as possible, we select three fonts, namely Gothic typefaces (黑体, hēitǐ) in both Simplified and Traditional Chinese, and Small Seal Script (小篆, xiǎozhuàn). The three fonts correspond to the three channels of the character images, whose size is set to 32×32 pixel.

3.4 Selective Modality Fusion Module

After applying the previously mentioned semantic, phonetic and graphic encoders, we get the representation vectors $\mathbf{H}^t$, $\mathbf{H}^a$ and $\mathbf{H}^v$ in textual, acoustic and visual modalities. To predict the final correct Chinese characters, we develop a selective modality fusion module to integrate these vectors in different modalities. This module fuses information in two levels, i.e., character-level and sentence-level.

First, for each modality, a selective gate unit is employed to control how much information can flow to the mixed multimodal representation. For example, if a character is misspelled due to its similar pronunciation to the correct one, then more information in the acoustic modality should flow into the mixed representation. The gate values are computed by a fully-connected layer followed by a sigmoid function. The inputs include the character representation of three modalities and the mean of the semantic encoder output $\mathbf{H}^t$ to capture the overall semantics of the input sentence. Formally, we denote the gate values for the textual, acoustic and visual modalities as g^t, g^a and g^v. The mixed multimodal representation $\tilde{h}_i$ of the i-th character is computed as follows:

$$
\begin{aligned}
\bar{h}^t &= \frac{1}{N}\sum_{i=1}^{N} h_i^t \\
g_i^t &= \sigma(\mathbf{W}^t \cdot [h_i^t, h_i^a, h_i^v, \bar{h}^t] + b^t) \\
g_i^a &= \sigma(\mathbf{W}^a \cdot [h_i^t, h_i^a, h_i^v, \bar{h}^t] + b^a) \\
g_i^v &= \sigma(\mathbf{W}^v \cdot [h_i^t, h_i^a, h_i^v, \bar{h}^t] + b^v) \\
\tilde{h}_i &= g_i^t \cdot h_i^t + g_i^a \cdot h_i^a + g_i^v \cdot h_i^v
\end{aligned}
\tag{4}
$$

where $\mathbf{W}^t$, $\mathbf{W}^a$, $\mathbf{W}^v$, b^t, b^a, b^v are learnable parameters, σ is the sigmoid function, and $[\cdot]$ means the concatenation of vectors.

Then, we apply the Transformer to fully learn the semantic, phonetic and visual information at the sentence-level. The mixed representations of all characters are packed together into $\mathbf{H}_0 = [\tilde{h}_1, \tilde{h}_2, ..., \tilde{h}_N]$, and the probability distribution $\hat{y}_i$ of what the i-th character should be is derived as:

$$
\begin{aligned}
\mathbf{H}_l &= \text{Transformer}_l(\mathbf{H}_{l-1}), l \in [1, L'] \\
\hat{y}_i &= \text{softmax}(\mathbf{W}^o h_i + b^o), h_i \in \mathbf{H}_{L'}
\end{aligned}
\tag{5}
$$

where L' is the number of Transformer layers, $\mathbf{W}^o$ and b^o are learnable parameters.

3.5 Acoustic and Visual Pretraining

While acoustic and visual information is essential to the CSC task, equally important is how to associate them with the correct character. In order to learn the acoustic-textual and visual-textual relationships, we propose to pretrain the phonetic and the graphic encoders.

For the phonetic encoder, we design an Input Method pretraining objective, that the encoder should recover the Chinese character sequence given the input pinyin sequence. This is what the Chinese input methods do. We add a linear layer on the top of the encoder to transform the hidden states to the probability distributions over the Chinese character vocabulary. We pretrain the phonetic encoder with the pinyin of the sentences with spelling errors in the training data, and make it recover the character sequences without spelling errors.

For the graphic encoder, we design an Optical Character Recognition (OCR) pretraining objective. Given the Chinese character images, the graphic encoder learns the visual information to predict the corresponding character over the Chinese character vocabulary. This is like what the OCR task does, but our recognition is only conducted on the character level and typed scripts. During the pretraining, we also add a linear layer on the top to perform the classification.

Finally, we load the pretrained weights of the semantic encoder, phonetic encoder, and graphic encoder, and conduct the final training process with the CSC training data.

Training Set	#Sent	Avg. Length	#Errors
SIGHAN13	700	41.8	343
SIGHAN14	3,437	49.6	5,122
SIGHAN15	2,338	31.3	3,037
Wang271K	271,329	42.6	381,962
Total	277,804	42.6	390464
Test Set	**#Sent**	**Avg. Length**	**#Errors**
SIGHAN13	1,000	74.3	1,224
SIGHAN14	1,062	50.0	771
SIGHAN15	1,100	30.6	703
Total	3,162	50.9	2,698

Table 2: Statistics of the used datasets. All the training data are merged to train the REALISE model. The test sets are used separately to evaluate the model performance.

4 Experiments

In this section, we introduce experimental details and results on the SIGHAN benchmarks (Wu et al., 2013; Yu et al., 2014; Tseng et al., 2015). We then verify the effectiveness of our model by conducting ablation studies and analyses.

4.1 Data and Metrics

Following previous works (Wang et al., 2019; Cheng et al., 2020), we use the SIGHAN training data and the generated pseudo data (Wang et al., 2018, denoted as Wang271K) as the training set. We evaluate our model on the SIGHAN test sets

Dataset	Method	Detection Level				Correction Level			
		Acc	Pre	Rec	F1	Acc	Pre	Rec	F1
SIGHAN13	Sequence Labeling (Wang et al., 2018)	-	54.0	69.3	60.7	-	-	-	52.1
	FASpell (Hong et al., 2019)	63.1	76.2	63.2	69.1	60.5	73.1	60.5	66.2
	BERT (Cheng et al., 2020)	-	79.0	72.8	75.8	-	77.7	71.6	74.6
	SpellGCN (Cheng et al., 2020)	-	80.1	74.4	77.2	-	78.3	72.7	75.4
	SpellGCN† (Our reimplementation)	78.8	85.7	78.8	82.1	77.8	84.6	77.8	81.0
	BERT†	77.0	85.0	77.0	80.8	75.2	83.0	75.2	78.9
	REALISE†	**82.7**	**88.6**	**82.5**	**85.4**	**81.4**	**87.2**	**81.2**	**84.1**
SIGHAN14	Sequence Labeling (Wang et al., 2018)	-	51.9	66.2	58.2	-	-	-	56.1
	FASpell (Hong et al., 2019)	70.0	61.0	53.5	57.0	69.3	59.4	52.0	55.4
	SpellGCN (Cheng et al., 2020)	-	65.1	69.5	67.2	-	63.1	67.2	65.3
	BERT	75.7	64.5	68.6	66.5	74.6	62.4	66.3	64.3
	REALISE	**78.4**	**67.8**	**71.5**	**69.6**	**77.7**	**66.3**	**70.0**	**68.1**
SIGHAN15	KUAS (Chang et al., 2015)	53.2	57.5	24.6	34.4	51.5	53.7	21.1	30.3
	NTOU (Chu and Lin, 2015)	42.2	42.2	41.8	42.0	39.0	38.1	35.2	36.6
	NCTU-NTUT (Wang and Liao, 2015)	60.1	71.7	33.6	45.7	56.4	66.3	26.1	37.5
	HanSpeller++ (Zhang et al., 2015)	70.1	80.3	53.3	64.0	69.2	79.7	51.5	62.5
	LMC (Xie et al., 2015)	54.6	63.8	21.5	32.1	52.3	57.9	16.7	26.0
	Sequence Labeling (Wang et al., 2018)	-	56.6	69.4	62.3	-	-	-	57.1
	FASpell (Hong et al., 2019)	74.2	67.6	60.0	63.5	73.7	66.6	59.1	62.6
	Soft-Masked BERT (Zhang et al., 2020)	80.9	73.7	73.2	73.5	77.4	66.7	66.2	66.4
	SpellGCN (Cheng et al., 2020)	-	74.8	80.7	77.7	-	72.1	77.7	75.9
	BERT	82.4	74.2	78.0	76.1	81.0	71.6	75.3	73.4
	REALISE	**84.7**	**77.3**	**81.3**	**79.3**	**84.0**	**75.9**	**79.9**	**77.8**

Table 3: The performance of our model and all baseline models on SIGHAN test sets. The "†" symbol means we apply post-processing (Section 4.2) to the model outputs on SIGHAN13. Results of REALISE on all SIGHAN test sets outperforms all the corresponding baselines with a significance level $p < 0.05$.

in 2013, 2014 and 2015 (denoted as SIGHAN13, SIGHAN14 and SIGHAN15). Table 2 shows the data statistics. Originally, the SIGHAN datasets are in the Traditional Chinese. Following previous works (Wang et al., 2019; Cheng et al., 2020; Zhang et al., 2020), we convert them to the Simplified Chinese using the OpenCC tool[2].

Results are reported at the detection level and the correction level. At the detection level, a sentence is considered to be correct if and only if all the spelling errors in the sentence are detected successfully. At the correction level, the model must not only detect but also correct all the erroneous characters to the right ones. We report the accuracy, precision, recall and F1 scores on both levels.

4.2 Implementation Details

The REALISE model is implemented using PyTorch framework (Paszke et al., 2019) with the Transformer library (Wolf et al., 2020). The architecture of the semantic encoder is same as the $\text{BERT}_{\text{BASE}}$ (Devlin et al., 2019) model (i.e. 12 transformer layers with 12 attention heads, hidden size of 768). We initialize the semantic encoder with the weights of BERT-wwm model (Cui et al., 2019). For the phonetic sentence-level encoder, we set the number of layers to 4, and initialize its position embedding with BERT's position embedding. The selective modality fusion module has 3 transformer layers, i.e., $L' = 3$, and the prediction matrix $\mathbf{W}^o$ is tied with the word embedding matrix of the semantic encoder. All the embeddings and hidden states have the dimension of 768. We use the Pillow library to extract the Chinese character images. When processing the special tokens (e.g. `[CLS]` and `[SEP]` of BERT), we use the tensor with all zero values as their image inputs. We train our REALISE model with the AdamW (Loshchilov and Hutter, 2017) optimizer for 10 epochs. The learning rate is set to 5e-5, the batch size is set to 32, and the model is trained with learning rate warming up and linear decay.

In the SIGHAN13 test set, the annotation quality is relatively poor, that quite a lot of the mixed usage of auxiliary "的", "地", and "得" are not annotated (Cheng et al., 2020). Therefore, a well-performed model may obtain bad scores on it. To alleviate the problem, Cheng et al. (2020) proposes

[2] https://github.com/BYVoid/OpenCC

to continue finetuning the model on the SIGHAN13 training set before testing. We argue that it's not a good practice because it reduces the model performance. Instead, we use a simple and effective post-processing method. We simply remove all the detected and corrected "的", "地", and "得" characters from the model output and then evaluate with the ground truth of SIGHAN13 test set.

4.3 Baselines

We compare REALISE with the following baselines: **KUAS** (Chang et al., 2015), **NTOU** (Chu and Lin, 2015), **NCTU-NTUT** (Wang and Liao, 2015), **HanSpeller++** (Zhang et al., 2015), **LMC** (Xie et al., 2015) mainly utilize heuristics or traditional machine learning algorithms, such as n-gram language model, Conditional Random Field and Hidden Markov Model. **Sequence Labeling** (Wang et al., 2018) treats CSC as a sequence labeling problem and applies a BiLSTM model. **FASpell** (Hong et al., 2019) utilizes a denoising autoencoder (DAE) to generate candidate characters. **Soft-Masked BERT** (Zhang et al., 2020) utilizes the detection model to help the correction model learn the right context. **SpellGCN** (Cheng et al., 2020) incorporates the predefined character confusion sets to the BERT-based correction model through Graph Convolutional Networks (GCNs). **BERT** (Devlin et al., 2019) is to directly fine-tune the $\text{BERT}_{\text{BASE}}$ model with the CSC training data.

4.4 Main Results

Table 3 shows the evaluation scores at detection and correction levels on the SIGHAN 13/14/15 test sets. The REALISE model performs significantly better than all the previous state-of-the-art models on all test sets. It can be seen that, by capturing valuable information from acoustic and visual modalities, REALISE yields consistent gain with a large margin against BERT. Specifically, at the correction-level, REALISE exceeds BERT by 5.2% F1 on SIGHAN13, 3.8% F1 on SIGHAN14, and 4.4% F1 on SIGHAN15. The results on SIGHAN13 are improved significantly with simple post-processing described in Section 4.2.

There are several successful applications of BERT on the CSC task, such as FASpell and SpellGCN, which also consider the Chinese character similarity. They attempt to calculate the similarity as the confidence of filtering candidates, or construct similarity graphs from predefined confusion sets. Instead, in our method, multiple encoders are directly applied to derive more informative representation from the acoustic and visual modalities. Compared with SpellGCN (Cheng et al., 2020), the SOTA CSC model, our REALISE model achieves an averaging 2.4% F1 improvements at detection-level and an averaging 2.6% F1 improvements at correction-level. This indicates that, compared with other extensions of BERT, the explicit utilization of multimodal information of Chinese characters is more beneficial to the CSC task.

With the simple post-processing as described in Section 4.2, results of each model on the SIGHAN13 test set are improved significantly. Compared with BERT and SpellGCN, we can see that, after the post-processing, the REALISE model is ahead of all the baseline models.

Model	Acc	Pre	Rec	F1
	Detection Level			
BERT	78.4	74.6	74.5	74.5
REALISE	82.0	77.9	78.5	78.1
- Phonetic	81.2	76.4	77.7	77.0
- Graphic	81.4	77.3	77.2	77.2
- Multi-Fonts	81.2	76.3	77.9	77.0
- Pretraining	81.5	76.5	78.1	77.2
- Selective-Fusion	81.3	76.8	77.4	77.1
	Correction Level			
BERT	76.9	72.3	72.3	72.3
REALISE	81.0	76.5	77.0	76.7
- Phonetic	80.2	74.8	76.1	75.4
- Graphic	80.5	75.8	75.6	75.7
- Multi-Fonts	80.3	74.9	76.4	75.5
- Pretraining	80.6	75.2	76.8	75.9
- Selective-Fusion	80.5	75.4	76.0	75.7

Table 4: Ablation results of the REALISE model averaged on SIGHAN test sets. We apply the following changes to REALISE: removing the phonetic encoder (- Phonetic), removing the graphic encoder (- Graphic), using only one font to build the graphic inputs (- Multi-Fonts), removing acoustic and visual pretraining (- Pretraining), replacing the selective modality fusion mechanism with simple summation (- Selective-Fusion).

4.5 Ablation Study

We explore the contribution of each component in REALISE by conducting ablation studies with the following settings: 1) removing the phonetic encoder, 2) removing the graphic encoder, 3) using only one font (Gothic typefaces in Simplified Chinese) for the graphic encoder, 4) removing the acoustic and visual pretraining objectives, 5) replacing the selective modality fusion mechanism with simple summation.

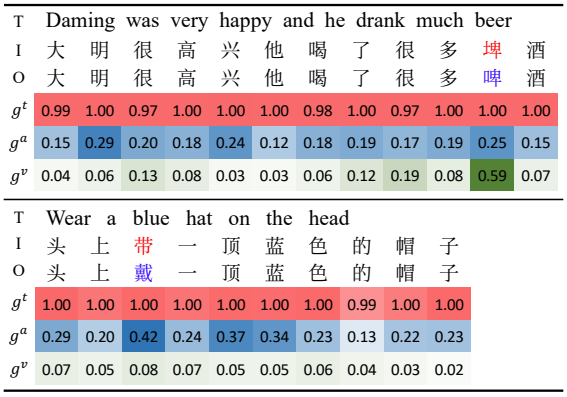

Figure 2: Selective modality fusion visualization. "I" is the input sentence. "O" is the output of REALISE (also the ground truth), and "T" is the translation. g^t, g^a, g^v are the gate values for the textual, acoustic, and visual modality respectively. We highlight the wrong/correct characters in red/blue color.

Table 4 shows the averaged scores[3] on three SIGHAN test sets. The main motivation of this paper is to discover the character similarity relationships by incorporating the acoustic and visual information. If removing the phonetic or graphic encoder, we can see that the model performance drops at two levels but still outperforms BERT significantly. This suggests that the checking model can benefit from the multimodal information. No matter which component we remove, the performance of REALISE drops, which fully demonstrates the effectiveness of each part in our model.

4.6 Analysis of the Selective Modality Fusion Module

Figure 2 gives two examples to analyze the selective modality fusion module. In the first example, the acoustic and visual selective gate values of "埤", i.e. g^a and g^v, are much larger than most other characters since "埤(pí)" and "啤(pí)" have the same pronunciation and right radical "卑". This shows that the selective fusion module can judge whether to introduce phonetic or graphic information into the mixed representation. The second example shows a similar trend for the pronunciation of "带(dài)" and "戴(dài)". More selective fusion visualization can be found in the Appendix A.2.

Besides, we calculate the averaged gate values of erroneous characters for each modality on SIGHAN15. The largest one is the textual modality that the value is almost equal to 1.0. The second one is the acoustic modality that the averaged value is 0.334, and the smallest one is the visual modality that the value is 0.229. It means that the information from the semantic encoder is the most important for correcting the spelling errors. The acoustic modality is more important than the visual modality, which is consistent with the fact that the spelling errors caused by similar pronunciations are more frequent than errors caused by similar character shapes (Liu et al., 2010).

[3]Full ablation results can be found in the Appendix A.1.

In:	我打算去法国流行，你要不要跟我一起去? I am going to popular to France, would you like to go with me?
Out:	我打算去法国旅行，你要不要跟我一起去? I am going to travel to France, would you like to go with me?
In:	回国之后，我跟快去你家。 After returning home, I will go to your house with.
Out:	回国之后，我很快去你家。 After returning home, I will go to your house soon.

Table 5: Examples of the input and output of our REALISE model. We highlight the wrong/correct characters in red/blue color.

4.7 Case Study

In the first example in Table 5, "流" is the erroneous character. If ignoring the Chinese character similarities, we can find that there are multiple candidate corrections to replace the "流" character. For instance, we can replace it with "游" and the English translation is "*I am going to parade in France*". However, the REALISE's output is the best correction, because "流(liú)" and "旅(lˇü)" have a similar pronunciation. In the second example, not only the phonetic information, but also the visual information is important for correcting "跟(gēn)" to "很(hěn)". In detail, the two characters share the same final "en" in pronunciation, and have the same right radical "艮".

The errors in the above examples are not corrected by SpellGCN, since they are not defined as confusing character pairs in the handcrafted confusion sets (Lee et al., 2019). Specifically, in the SIGHAN15 test set, there are 16% erroneous-corrected character pairs not in the predefined confusion sets. SpellGCN corrects 64.6% of them but REALISE performs better with 73.5% correction. Besides, for the easily-confused pairs in the predefined sets, SpellGCN corrects 82.5% of them and REALISE corrects 85.8%. This indicates that leveraging multimodal information of Chinese charac-

ters helps the model generalize better in capturing the character similarity relationships.

5 Conclusion

In this paper, we propose a model called REALISE for Chinese spell checking. Since the spelling errors in Chinese are often semantically, phonetically or graphically similar to the correct characters, REALISE leverages information in textual, acoustic and visual modalities to detect and correct the errors. The REALISE model captures information in these modalities using tailored semantic, phonetic and graphic encoders. Besides, a selective modality fusion mechanism is proposed to control the information flow of these modalities. Experiments on the SIGHAN benchmarks show that the proposed REALISE outperforms the baseline models using only textual information by a large margin, which verifies that leveraging acoustic and visual information helps the Chinese spell checking task.

References

Haithem Afli, Zhengwei Qiu, Andy Way, and Páraic Sheridan. 2016. Using smt for ocr error correction of historical texts. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 962–966.

Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C. Lawrence Zitnick, and Devi Parikh. 2015. VQA: visual question answering. In *2015 IEEE International Conference on Computer Vision, ICCV 2015, Santiago, Chile, December 7-13, 2015*, pages 2425–2433. IEEE Computer Society.

Jimmy Ba, Jamie Ryan Kiros, and Geoffrey E. Hinton. 2016. Layer normalization. *ArXiv*, abs/1607.06450.

Loïc Barrault, Fethi Bougares, Lucia Specia, Chiraag Lala, Desmond Elliott, and Stella Frank. 2018. Findings of the third shared task on multimodal machine translation. In *Proceedings of the Third Conference on Machine Translation: Shared Task Papers, WMT 2018, Belgium, Brussels, October 31 - November 1, 2018*, pages 304–323. Association for Computational Linguistics.

Tao-Hsing Chang, Hsueh-Chih Chen, and Cheng-Han Yang. 2015. Introduction to a proofreading tool for chinese spelling check task of SIGHAN-8. In *Proceedings of the Eighth SIGHAN Workshop on Chinese Language Processing, SIGHAN@IJCNLP 2015, Beijing, China, July 30-31, 2015*, pages 50–55. Association for Computational Linguistics.

Wei-Lun Chao, Hexiang Hu, and Fei Sha. 2018. Being negative but constructively: Lessons learnt from creating better visual question answering datasets. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2018, New Orleans, Louisiana, USA, June 1-6, 2018, Volume 1 (Long Papers)*, pages 431–441. Association for Computational Linguistics.

Xingyi Cheng, Weidi Xu, Kunlong Chen, Shaohua Jiang, Feng Wang, Taifeng Wang, Wei Chu, and Yuan Qi. 2020. Spellgcn: Incorporating phonological and visual similarities into language models for chinese spelling check. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 871–881. Association for Computational Linguistics.

Kyunghyun Cho, Bart van Merrienboer, Çaglar Gülçehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using RNN encoder-decoder for statistical machine translation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP 2014, October 25-29, 2014, Doha, Qatar, A meeting of SIGDAT, a Special Interest Group of the ACL*, pages 1724–1734. ACL.

Wei-Cheng Chu and Chuan-Jie Lin. 2015. NTOU chinese spelling check system in sighan-8 bake-off. In *Proceedings of the Eighth SIGHAN Workshop on Chinese Language Processing, SIGHAN@IJCNLP 2015, Beijing, China, July 30-31, 2015*, pages 137–143. Association for Computational Linguistics.

Yiming Cui, Wanxiang Che, Ting Liu, Bing Qin, Ziqing Yang, Shijin Wang, and Guoping Hu. 2019. Pre-training with whole word masking for chinese BERT. *CoRR*, abs/1906.08101.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. 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, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 4171–4186. Association for Computational Linguistics.

Fei Dong and Yue Zhang. 2016. Automatic features for essay scoring–an empirical study. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1072–1077.

Li Dong, Nan Yang, Wenhui Wang, Furu Wei, Xiaodong Liu, Yu Wang, Jianfeng Gao, Ming Zhou, and Hsiao-Wuen Hon. 2019. Unified language model pre-training for natural language understanding and generation. In *33rd Conference on Neural Information Processing Systems (NeurIPS 2019)*.

Jianfeng Gao, Xiaolong Li, Daniel Micol, Chris Quirk, and Xu Sun. 2010. A large scale ranker-based sys-

tem for search query spelling correction. In *Proceedings of the 23rd International Conference on Computational Linguistics (Coling 2010)*, pages 358–366, Beijing, China. Coling 2010 Organizing Committee.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, pages 770–778. IEEE Computer Society.

Julian Hitschler, Shigehiko Schamoni, and Stefan Riezler. 2016. Multimodal pivots for image caption translation. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016, August 7-12, 2016, Berlin, Germany, Volume 1: Long Papers*. The Association for Computer Linguistics.

Yuzhong Hong, Xianguo Yu, Neng He, Nan Liu, and Junhui Liu. 2019. 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@EMNLP 2019, Hong Kong, China, November 4, 2019*, pages 160–169. Association for Computational Linguistics.

Norman Jerry. 1988. Chinese (cambridge language surveys).

Thomas N Kipf and Max Welling. 2016. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*.

Lung Hao Lee, Wun Syuan Wu, Jian Hong Li, Yu Chi Lin, and Yuen Hsien Tseng. 2019. Building a confused character set for chinese spell checking. In *27th International Conference on Computers in Education, ICCE 2019*, pages 703–705. Asia-Pacific Society for Computers in Education.

Gen Li, Nan Duan, Yuejian Fang, Ming Gong, and Daxin Jiang. 2020. Unicoder-vl: A universal encoder for vision and language by cross-modal pre-training. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pages 11336–11344. AAAI Press.

C.-L. Liu, M.-H. Lai, K.-W. Tien, Y.-H. Chuang, S.-H. Wu, and C.-Y. Lee. 2011. Visually and phonologically similar characters in incorrect chinese words: Analyses, identification, and applications. *ACM Transactions on Asian Language Information Processing*, 10(2).

Chao-Lin Liu, Min-Hua Lai, Yi-Hsuan Chuang, and Chia-Ying Lee. 2010. Visually and phonologically similar characters in incorrect simplified Chinese words. In *Coling 2010: Posters*, pages 739–747, Beijing, China. Coling 2010 Organizing Committee.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. RoBERTa: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Ilya Loshchilov and Frank Hutter. 2017. Fixing weight decay regularization in adam. *CoRR*, abs/1711.05101.

Bruno Martins and Mário J. Silva. 2004. Spelling correction for search engine queries. In *Advances in Natural Language Processing*, pages 372–383, Berlin, Heidelberg. Springer Berlin Heidelberg.

Yuxian Meng, Wei Wu, Fei Wang, Xiaoya Li, Ping Nie, Fan Yin, Muyu Li, Qinghong Han, Xiaofei Sun, and Jiwei Li. 2019. Glyce: Glyph-vectors for chinese character representations. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, 8-14 December 2019, Vancouver, BC, Canada*, pages 2742–2753.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Köpf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. 2019. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 8024–8035.

Weijie Su, Xizhou Zhu, Yue Cao, Bin Li, Lewei Lu, Furu Wei, and Jifeng Dai. 2020. VL-BERT: pre-training of generic visual-linguistic representations. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

Yu Sun, Shuohuan Wang, Yu-Kun Li, Shikun Feng, Hao Tian, Hua Wu, and Haifeng Wang. 2020. ERNIE 2.0: A continual pre-training framework for language understanding. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pages 8968–8975. AAAI Press.

Ryuichi Tachibana and Mamoru Komachi. 2016. Analysis of english spelling errors in a word-typing game. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 385–390.

Hao Tan and Mohit Bansal. 2019. LXMERT: learning cross-modality encoder representations from transformers. In *Proceedings of the 2019 Conference on*

Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019, pages 5099–5110. Association for Computational Linguistics.

Yuen-Hsien Tseng, Lung-Hao Lee, Li-Ping Chang, and Hsin-Hsi Chen. 2015. Introduction to SIGHAN 2015 bake-off for chinese spelling check. In *Proceedings of the Eighth SIGHAN Workshop on Chinese Language Processing, SIGHAN@IJCNLP 2015, Beijing, China, July 30-31, 2015*, pages 32–37. Association for Computational Linguistics.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems*, 30:5998–6008.

Oriol Vinyals, Meire Fortunato, and Navdeep Jaitly. 2015. Pointer networks. *Advances in neural information processing systems*, 28:2692–2700.

Dingmin Wang, Yan Song, Jing Li, Jialong Han, and Haisong Zhang. 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, Brussels, Belgium, October 31 - November 4, 2018*, pages 2517–2527. Association for Computational Linguistics.

Dingmin Wang, Yi Tay, and Li Zhong. 2019. Confusionset-guided pointer networks for chinese spelling check. In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pages 5780–5785. Association for Computational Linguistics.

Yih-Ru Wang and Yuan-Fu Liao. 2015. Word vector/conditional random field-based chinese spelling error detection for SIGHAN-2015 evaluation. In *Proceedings of the Eighth SIGHAN Workshop on Chinese Language Processing, SIGHAN@IJCNLP 2015, Beijing, China, July 30-31, 2015*, pages 46–49. Association for Computational Linguistics.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

Shih-Hung Wu, Chao-Lin Liu, and Lung-Hao Lee. 2013. Chinese spelling check evaluation at SIGHAN bake-off 2013. In *Proceedings of the Seventh SIGHAN Workshop on Chinese Language Processing, SIGHAN@IJCNLP 2013, Nagoya, Japan, October 14-18, 2013*, pages 35–42. Asian Federation of Natural Language Processing.

Weijian Xie, Peijie Huang, Xinrui Zhang, Kaiduo Hong, Qiang Huang, Bingzhou Chen, and Lei Huang. 2015. Chinese spelling check system based on n-gram model. In *Proceedings of the Eighth SIGHAN Workshop on Chinese Language Processing, SIGHAN@IJCNLP 2015, Beijing, China, July 30-31, 2015*, pages 128–136. Association for Computational Linguistics.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Ruslan Salakhutdinov, and Quoc V Le. 2019. XLNet: Generalized autoregressive pretraining for language understanding. *arXiv preprint arXiv:1906.08237*.

Junjie Yu and Zhenghua Li. 2014. 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*, pages 220–223.

Liang-Chih Yu, Lung-Hao Lee, Yuen-Hsien Tseng, and Hsin-Hsi Chen. 2014. Overview of SIGHAN 2014 bake-off for chinese spelling check. In *Proceedings of The Third CIPS-SIGHAN Joint Conference on Chinese Language Processing, Wuhan, China, October 20-21, 2014*, pages 126–132. Association for Computational Linguistics.

Amir Zadeh, Rowan Zellers, Eli Pincus, and Louis-Philippe Morency. 2016. Multimodal sentiment intensity analysis in videos: Facial gestures and verbal messages. *IEEE Intell. Syst.*, 31(6):82–88.

Dong Zhang, Shoushan Li, Qiaoming Zhu, and Guodong Zhou. 2019. Effective sentiment-relevant word selection for multi-modal sentiment analysis in spoken language. In *Proceedings of the 27th ACM International Conference on Multimedia*, pages 148–156.

Shaohua Zhang, Haoran Huang, Jicong Liu, and Hang Li. 2020. Spelling error correction with soft-masked BERT. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 882–890. Association for Computational Linguistics.

Shuiyuan Zhang, Jinhua Xiong, Jianpeng Hou, Qiao Zhang, and Xueqi Cheng. 2015. Hanspeller++: A unified framework for chinese spelling correction. In *Proceedings of the Eighth SIGHAN Workshop on Chinese Language Processing, SIGHAN@IJCNLP 2015, Beijing, China, July 30-31, 2015*, pages 38–45. Association for Computational Linguistics.

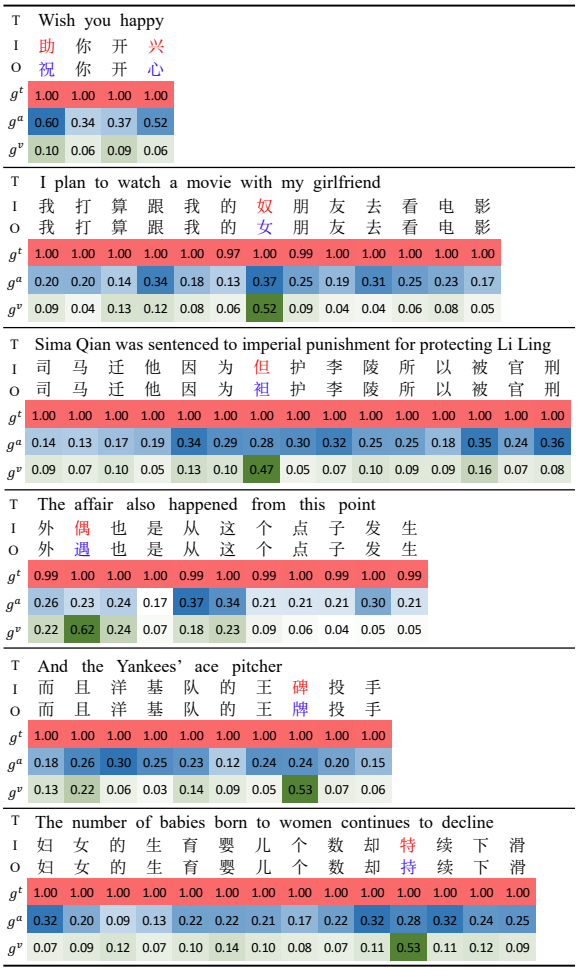

Figure 3: Selective modality fusion visualization. "I" is the input sentence. "O" is the output of REALISE (also the ground truth), and "T" is the translation. g^t, g^a, g^v are the gate values for the textual, acoustic, and visual modality respectively. We highlight the wrong/correct characters in red/blue color.

A Appendix

A.1 Ablation

We conduct an ablation study to verify the effectiveness of the proposed method. Table 6 (on Page 13 in Appendix) shows the detailed ablation results on each SIGHAN test set, where the following settings are conducted:

1. **- Phonetic**: removing the phonetic encoder.
2. **- Graphic**: removing the graphic encoder.
3. **- Multi-Fonts**: using only one font (Gothic typefaces in Simplified Chinese) for the graphic encoder.
4. **- Pretraining**: removing the acoustic and visual pretraining objectives.
5. **- Selective-Fusion**: replacing the selective modality fusion mechanism with simple summation.

We can see that, when we remove anything from our model, the REALISE performance drops consistently, and it drops most apparently in the SIGHAN14 test set. These results suggest that each part of our model is an effective means for boosting the checking performance.

A.2 Selective Modality Fusion Visualization

We show more examples in Figure 3. We can see that, if the misused characters are phonetically similar to the correct ones, the acoustic gate values tend to be larger, and if they are graphically similar, the visual gate values are larger.

Dataset	Method	Detection Level				Correction Level			
		Acc	Pre	Rec	F1	Acc	Pre	Rec	F1
SIGHAN13	BERT	77.0	85.0	77.0	80.8	75.2	83.0	75.2	78.9
	REALISE	82.7	88.6	82.5	85.4	81.4	87.2	81.2	84.1
	- Phonetic	82.4	87.4	82.3	84.8	81.2	86.1	81.1	83.5
	- Graphic	82.1	88.1	82.1	85.0	80.9	86.7	80.8	83.7
	- Multi-Fonts	82.2	87.5	82.2	84.8	81.2	86.4	81.2	83.7
	- Pretraining	82.8	88.2	82.7	85.4	81.4	86.7	81.3	83.9
	- Selective-Fusion	82.0	87.3	82.0	84.6	81.0	86.2	81.0	83.5
SIGHAN14	BERT	75.7	64.5	68.6	66.5	74.6	62.4	66.3	64.3
	REALISE	78.4	67.8	71.5	69.6	77.7	66.3	70.0	68.1
	- Phonetic	77.1	65.5	69.2	67.3	76.3	63.8	67.5	65.6
	- Graphic	78.0	67.3	69.6	68.4	77.1	65.6	67.9	66.7
	- Multi-Fonts	76.9	65.0	69.6	67.2	76.2	63.6	68.1	65.7
	- Pretraining	77.5	65.6	70.4	67.9	76.7	64.0	68.7	66.2
	- Selective-Fusion	77.6	66.5	69.0	67.7	76.9	64.8	67.3	66.0
SIGHAN15	BERT	82.4	74.2	78.0	76.1	81.0	71.6	75.3	73.4
	REALISE	84.7	77.3	81.3	79.3	84.0	75.9	79.9	77.8
	- Phonetic	84.2	76.2	81.7	78.9	83.3	74.5	79.9	77.1
	- Graphic	84.3	76.6	79.9	78.2	83.5	75.0	78.2	76.6
	- Multi-Fonts	84.5	76.5	81.9	79.1	83.5	74.6	79.9	77.1
	- Pretraining	84.2	75.7	81.3	78.4	83.7	74.9	80.4	77.5
	- Selective-Fusion	84.4	76.8	81.2	78.9	83.6	75.4	79.7	77.5

Table 6: Ablation results of the REALISE model on each SIGHAN dataset.