Chinese Event Extraction from Handwritten Image

Anonymous ACL submission

Abstract

Event extraction is crucial for distilling semantic information from sentences, has traditionally focused on text-based input. However, as a long-existing data source for the needs of rapid recording, handwriting somehow has long been neglected in prior works. In this study, we move our sight to a more practical setting that aims to directly extract downstream event information from handwritten images. Under this setting, we first construct a benchmark dataset with manual annotation of handwritten images. This dataset is more challenging due to the widely distributed corner cases in handwritten images such as typos and scribbled writing styles, not even mention the serious cross-modality error propagation. We further propose Chinese Handwriting Vision-Language Model (HVLM) that consists of three joint training subtasks targeted at the challenging root of this setting. By emulating human reading habits, our model can quickly scan and precisely locate key information within the image, thereby improving the overall performance of extraction. Experiments demonstrate the huge advantage of our proposed model over the cutting-edge baselines, underscoring the necessity of introducing this new setting thereby guiding the holistic optimization on this real-world challenges.

1 Introduction

Event extraction aims to extract event records from the sentence, each of which includes four types of elements: a *trigger* and multiple *arguments* exist as the raw spans in the sentence, an *event type* or *role type* are assigned to corresponding trigger or argument as a result of classification.

Recent event extraction works on Chinese show a similar routine to English, have experienced a migration of methods shifted from the classification methods (Sha et al., 2016) to generative methods (Lu et al., 2021), but tailored to accommodate unique characteristics of the Chinese language (Li and Zhou, 2012), such as word-based feature (Ding et al., 2019), speech (Zhang et al., 2023) and glyph (Bao et al., 2024), achieved promising improvement in performance.

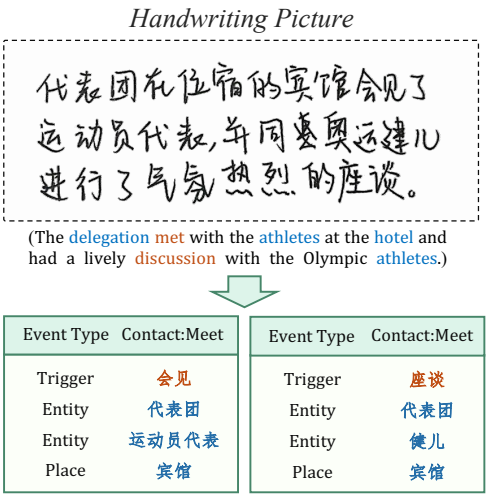


Figure 1: The illustration of Chinese event extraction from handwritten image.

However, a more practical setting of data source: handwritten images, have been ignored for a long time. Handwritten images not only serve as the form in which many textual dataset exist in the real world, but also emerges as the more direct medium for rapidly conveying event information, such as stenography and instrument (Yang et al., 2019; Yu et al., 2020). Thus, event extraction from handwriting would exhibit increasing interest.

In this study, we move our sight to a new setting of *event extraction from handwritten image*. As shown in Figure 1, different from the common event extraction from texts, the input of our new setting is the unprocessed handwritten images, and the output target is the events exist in the context of the handwriting. We select Chinese due to its unique script, which lacks clear word boundaries compared to alphabetical scripts. Moreover,

Chinese word identification relies heavily on the arrangement of radicals, even slight mispositioning can lead to incorrect semantics interpretations. Our setting focuses on the most practical situation, thereby dominantly guiding scholars' attention to and solving this real-world dilemma.

In this paper, we construct a dataset named Chinese Handwriting Event Extraction (CHW-EE). On the basis of event labels from ACE2005 Chinese dataset, we build the testset by hiring naive speakers to handwrite the sentences sampled from the original ACE (Walker et al., 2006) dataset. Our annotation is designed to simulate real-life situations as closely as possible, the perspectives include the crowdsourcing of different writing styles from neat to scribble, the variations of spaces, numbers or punctuations, and even the minor corrections or typos during writing. Thereby our CHW-EE can facilitate the exhaustive benchmark for the event extraction from handwritten image.

However, it is difficult to extract event information from handwritten images. The very first challenge lies in the step of handwriting recognition since it serves as the basis of subsequent processing, it includes two aspects: 1) **Different Writing Styles** resulted from individual writing habits, proposing a hard challenge for the generalization of extraction model. 2) **Corrections and Typos** such as the "健" (strong) is misspelled into "建" (build) in Figure 1, which could lead to false recognition and semantic misunderstanding. The second challenge lies in 3) **Error Propagation** during the subsequent step of event extraction from the recognized context, where the input may contain false recognitions and even missing content that could result in serious semantic misunderstanding propagated from previous recognition.

We further present the first work to tackle event extraction from handwriting in an end-to-end way: Handwriting Vision-Language Model (HVLM). We address the above challenges mentioned with Reading Joint Training(RJT), it mimics the human procedure of reading Chinese handwriting with three subtasks: 1) Font Augmentation that targeted at reinforcing the model's robustness and generalization towards different writing styles by exposing it to fonts vary from neat to scribble; 2) In-Context Text Recognition to simulate the human reading procedure of skimming and segmenting the entire context, thereby skipping and altering the typos or corrections that could impend the reading; 3) Event Object Detection borrowed from computer vision that mock the human's active searching and pinpointing the key information in the image to bridge the gap between two modalities involved, thereby alleviating the error propagation; We subsequently finetune our HVLM with RJT to facilitate the effective extraction.

We finally benchmark our dataset using our HVLM and a set of representative baselines. The empirical results highlight the advantages of HVLM and validate the effectiveness of our Reading Joint Training that directly targets the core challenges. The results also lead us to conclude that, for real-world scenarios, framing the problem as a unified task with a globally optimal design may be a more practical and effective choice.

2 Related Work

Event extraction works have indeed leveraged modeling methods from diverse perspectives, from the sequence tagging (Lin et al., 2020; Fan and He, 2023), structure and graph (Lin et al., 2020; Liu et al., 2023) to multi-modalities(Li et al., 2023b,a; Zhang et al., 2024a). Recent trends have shifted towards harnessing LLMs to generate the target sequence (Lu et al., 2021; Liu et al., 2023; Yang et al., 2023; Zhang et al., 2024b).

Few studies have developed event extraction methods specifically tailored to the unique linguistic and structural characteristics of Chinese, with most approaches adapting techniques originally designed for English datasets (Li and Zhou, 2012; Li et al., 2012; Ding et al., 2019). Earlier studies often relied on hand-crafted features and patterns, limiting compatibility with modern deep learning methods. Recent neural network-based approaches have made significant progress using raw inputs. For example, JMCEE (Xu et al., 2020) effectively addressed the challenge of overlapping roles, while PAJHEE (Shen et al., 2020) introduced hierarchical representations to capture event-level features. Additionally, NPN (Lin et al., 2018) utilized a character-level hybrid representation to improve event detection, and GVLM (Bao et al., 2024) further enhanced extraction by incorporating sentence-level glyphic information, leveraging the visual features of characters.

However, none of the above works considered handwritten images, which serves as a more practical data source close to reality when compared with the over-idealistic cleaned text-based data. To the best of our knowledge, our work stands out as

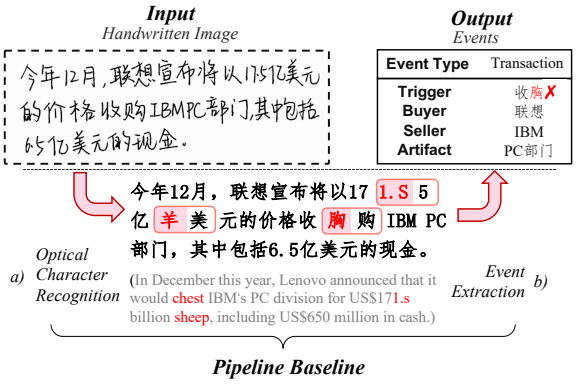


Figure 2: The workflow of pipeline baseline.

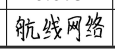
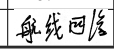
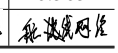
	Neat	Medium	Scribble
Samples	300	400	300
Avg. Edit Distance	0.037	0.076	0.181
Avg. Typo	0.13	0.21	0.44
Avg. Correction	0.06	0.14	0.29
Neatness Score	0.075	0.142	0.303
Example			

Table 1: Statistics of three neatness levels in our testset.

the first to perform the downstream event extraction from unprocessed handwritten images.

3 Preliminaries

3.1 Chinese Event Extraction from Handwritten Image

As shown in Figure 2, we first define the input of this setting is the unprocessed handwriting image solely without any textual context. The output will be the standard events consist of trigger and arguments along with the event and role types. The event extraction from handwritten image is then defined as a task to extract a set of events E described in the context of a handwritten image i containing n words context $i = [w_1, ..., w_n]$:

$$E = (t, e, \{a_1\text{-}r_1, a_2\text{-}r_2, \ldots, a_k\text{-}r_k\}) \quad (1)$$

where t is the rooted trigger, $e \in \mathcal{T}$ is its event type, a_1 is the extracted argument, while $r_1 \in \mathcal{R}$ is argument's role type. $\mathcal{T}$ and $\mathcal{R}$ are the pre-defined sets of event and role types respectively.

3.2 Dataset

We construct a new dataset called **C**hinese **H**and**W**riting **E**vent **E**xtraction **(CHW-EE)** for benchmarking. CHW-EE focuses on one of the most challenging hieroglyphic languages, Chinese, leveraging the event labels from the ACE2005 Chinese dataset (Walker et al., 2006).

We first combined the splits of the original Chinese ACE 2005 dataset and sample 1,000 sentences for the test set, the remained 6,914 sentences were split into a train set containing 6,000 samples and a dev set with 914 samples. For the train and dev set, we build the input images by printing each sample into an image of 1024 $\times$ 1024 with the font of Song (宋), composing of the characters' pixel maps that are concatenated with a common modern Chinese writing order: from left to right and starting a new line below current one.

For the test set, we sampled 1,000 sentences with lengths ranging between 50 and 70 words. To construct a diverse testset that could reflects real-world scenarios, we hired 10 native speakers to mock different levels of writing neatness. To quantitatively assess neatness, we designed metrics including: (1) average minimum edit distance per character between PaddleOCR (Du et al., 2021) output and ground truth, (2) average typos and corrections per sample, and (3) an overall neatness score (the average of the above), where lower scores indicate neater handwriting. Annotators with overall neatness scores ≥ 0.3 are labeled as scribbled, 0.1–0.3 as medium, and ≤ 0.1 as neat. We task each annotator to write 100 sentences under realistic conditions, including spaces, punctuation, and occasional language switches to reflect real-world handwriting. Minor corrections and typos were permitted, but excessive errors were discouraged with the requirement of no more than one correction or typo per sample in average.

We show the detailed statistics for the three neatness levels in Table 1. We can tell that three of the annotators are classified at the scribbled level, four at the medium level, and the remaining three at the neat level. The three levels have significant differences in the level of neatness as we show in both the statistics and cases, thereby can facilitate our subsequent analysis of how does the neatness level affect the extraction in Section 6.3.

3.3 Basic Pipeline Model and Challenges

We design a set of pipeline baselines for our task. Since the input and output of our task are from two modalities, these baselines that are formed in an intuitive two-step manner: 1) The input image will first been processed by the OCR models for recognizing the characters in the handwriting images and forming them into readable sentences

as shown in Figure 2 (a). 2) The event extraction model will accept the recognized sentence and simply deals with it as an all-good input like a traditional event extraction as shown in Figure 2 (b).

Based on the characteristics of our task, the challenges of our task lay in the two key steps towards the final extraction, the first is the ***handwriting recognition*** step, it includes two aspects:

- **Corrections & Typos**: Corrections occurred when there were errors during writing. Cross, slash and smudge are common notations for correction. We show the case of slash in Figure 1, where the general structure of the corrected characters still remained. Typos are the errors that have not been corrected. Even small corrections and typos could result in serious semantic misunderstanding for the model.

- **Different Writing Styles**: Different individuals will have significant variance in their writing styles. It could lay in the scripts of single character, from neat to scribble. It can also be located in the overall structure of the handwriting such as different row and column spacings.

The second challenge lies in the ***event extraction*** from the recognized context, compared with the regularly extraction from plain text, the major challenging point here can be summarized as:

- **Error Propagation**: as a cross-modal task, the final events are extracted from the recognized handwriting context, the errors occurred during the recognition would inevitably affect model's semantic understanding of the context, therefore been propagated to the final extraction. Such a cross-modal pattern requires the extraction step having strong robustness and generalization towards various erroneous recognition.

4 Handwriting Vision-Language Model

In this study, we introduce a novel Handwriting Vision-Language Model (HVLM). As illustrated in Figure 3, we tackle the challenges by designing three subtasks that mirror how humans address them: Event Object Detection to mitigate error propagation, In-Context Text Recognition to handle corrections and typos, and Font Augmentation to accommodate diverse writing styles. We then execute these three subtasks jointly within our HVLM. We will discuss these steps one by one.

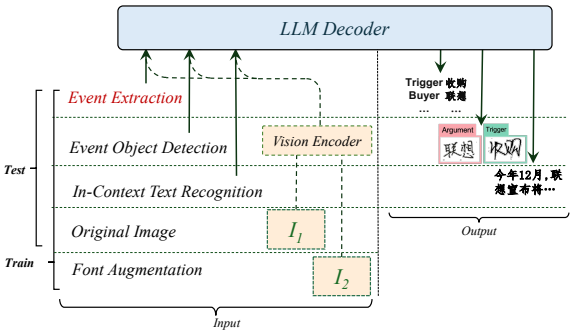


Figure 3: Our handwriting vision-language model.

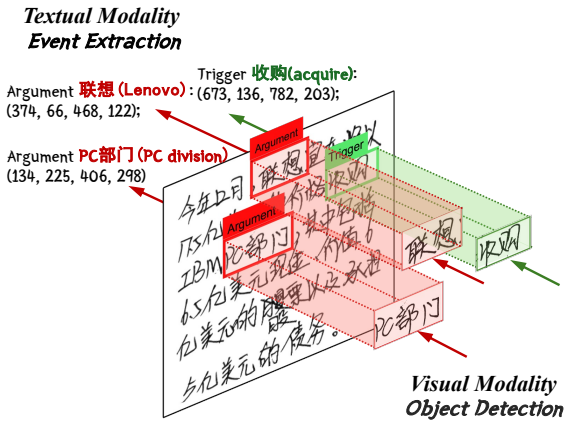


Figure 4: The illustration of our event object detection.

4.1 Event Object Detection

To address the challenge of error propagation, we are motivated to develop a method that connects the visual and textual modalities. Inspired by how humans read handwritten content, we note that the process of extracting textual information from image inputs mirrors how humans read, akin to object detection in computer vision: Readers tend to focus on salient details within localized regions and then interpret them in context, which aligns with the practice of identifying image bounding boxes in object detection, as shown in Figure 4.

We thus borrow this concept by projecting object detection to our handwritten images as event object detection, the target is the position of triggers and arguments that human would focus on when they are reading for extracting events. Particularly, given the handwritten image as input, different from the object detection in computer vision that performs classification upon pixels, our Event Object Detection cast the bounding box into target sequence: For a trigger T whose top left corner's coordinate is X_{left}^{top}, the target sequence of our Event Object Detection is the string composed

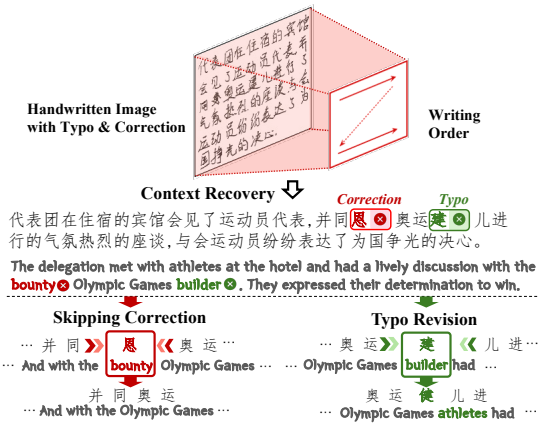


Figure 5: Our in-context text recognition.

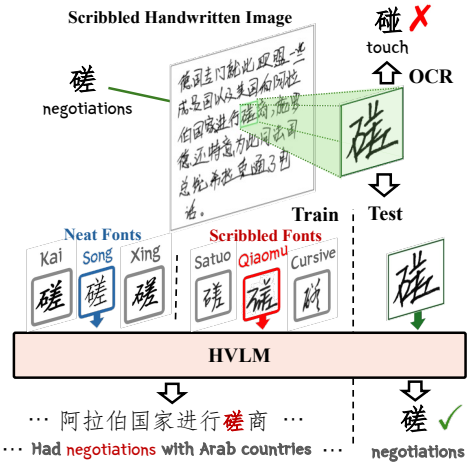


Figure 6: The illustration of our font augmentation.

of the coordinates of a bounding box b:

$$\text{b}= [\ (X^{top}_{left}, Y^{top}_{left}), (X^{bottom}_{right}, Y^{bottom}_{right})] \quad (2)$$

In our Event Object Detection, the decoder predicts this target sequence token-by-token. At the i-th step of generation, the decoder predicts the i-th token y_i in the output with decoder state h^d_i:

$$\text{y}_i, h^d_i = ([h^d_1, ..., h^d_{i-1}], y_{i-1}) \quad (3)$$

The conditional probability of the whole output sequence $p(y|x)$ is progressively combined by the probability of each step $p(y_i|y_{<i}, x)$:

$$\text{p(y|x)}= \prod_{i=1}^{|y|} p(y_i|y_{<i}, x) \quad (4)$$

where $y_{<i} = y_1...y_{i-1}$, and $p(y_i|y_{<i}, x)$ are the probabilities over target vocabulary V.

Our target string of bounding box is tokenized as text labels and does not require an additional positional vocabulary. By doing so, we can novelly connect text generation with vision through object detection and bridge the input-output modalities.

4.2 In-Context Text Recognition

We then tackle the issue of corrections and typos. A major limitation of OCRs is their character-by-character recognition, which prevents them from correcting even simple mistakes. In contrast, humans rely on in-context recognition, they can use the surrounding contents and writing order to overlook or adjust typos and corrections that might otherwise hinder the understanding, as illustrated in Figure 5. Accordingly, we develop the following strategies for In-Context Text Recognition:

- **Context Recovery** serves as the basis for the following in-context recognition. The model is trained to recognize the language, writing structure and each single character in the handwritten image to recover it into a readable sentence context to be processed by the following steps.

- **Typo Revision** addresses typos, which are often overlooked in OCR-based methods but can lead to semantic misunderstandings during information extraction step. To mitigate this problem, we employ in-context learning by leveraging the surrounding contextual information to correct potential typos, thereby enhancing the robustness of the overall extraction process.

- **Skipping Correction** is similar to previous strategy but have different target. It focuses on the correction such as cross or slash that can be interpreted as unrelated characters during recognition as shown in Table 7. To tackle this problem, we employ in-context learning by utilizing the surrounding textual cues to guide the VLM in skipping these corrections and seamlessly transitioning to the subsequent context.

By doing so, our Text Recognition module can leverage the richer contextual information to better handle corrections and typos exist in the context, while also substantially reducing the errors that would otherwise propagate from recognition into the subsequent event extraction stages.

4.3 Font Augmentation

How to deal with different writing styles become our next target. The model unable to cope the various writing styles well because it has only been exposed to the standard fonts such as Song(宋体),

Method		Tri-I			Tri-C			Arg-I			Arg-C		
		P.	R.	F1.	P.	R.	F1.	P.	R.	F1.	P.	R.	F1.
Pipeline													
PaddleOCR +	NPN	0.474	0.486	0.479	0.448	0.436	0.441	0.403	0.385	0.393	0.377	0.354	0.366
	TLNN	0.468	0.475	0. 471	0.434	0.442	0.437	0.396	0.378	0.386	0.379	0.347	0.362
	ONEIE	0.482	0.493	0.487	0.462	0.447	0.454	0.413	0.382	0.396	0.383	0.359	0.370
	DEGREE	0.473	0.478	0.475	0.456	0.439	0.447	0.407	0.373	0.389	0.371	0.363	0.366
	Qwen2.5-7B	0.585	0.573	0.579	0.558	0.546	0.552	0.484	0.471	0.477	0.471	0.455	0.463
	LLaMA3-8B	0.510	0.470	0.489	0.468	0.431	0.449	0.413	0.355	0.382	0.391	0.337	0.362
	ChatGLM4-9B	0.544	0.513	0.528	0.525	0.479	0.501	0.462	0.398	0.427	0.432	0.376	0.402
EasyOCR +	ONEIE	0.469	0.487	0.477	0.451	0.433	0.441	0.406	0.377	0.390	0.375	0.354	0.364
	Qwen2.5-7B	0.566	0.574	0.568	0.546	0.538	0.541	0.452	0.463	0.457	0.426	0.432	0.428
	LLaMA3-8B	0.497	0.458	0.476	0.443	0.421	0.431	0.402	0.347	0.372	0.384	0.336	0.358
GOT-OCR2.0 +	ONEIE	0.464	0.482	0.472	0.446	0.439	0.442	0.398	0.380	0.388	0.372	0.356	0.363
	Qwen2.5-7B	0.562	0.578	0.569	0.541	0.535	0.537	0.446	0.467	0.456	0.429	0.425	0.426
	LLaMA3-8B	0.492	0.459	0.474	0.446	0.413	0.428	0.404	0.342	0.370	0.389	0.331	0.357
End-to-End													
QWen-2-VL		0.642	0.484	0.552	0.604	0.455	0.519	0.538	0.395	0.455	0.519	0.375	0.435
InternLM2-XComposer		0.562	0.462	0.506	0.524	0.433	0.474	0.449	0.335	0.383	0.442	0.327	0.376
Ours		**0.695**	**0.665**	**0.680**	**0.663**	**0.635**	**0.648**	**0.604**	**0.540**	**0.570**	**0.588**	**0.522**	**0.553**

Table 2: Comparison with baselines.

which although covers the clear structure of each character but lack the generalization towards variants such as consecutive or overlapped radicals.

Building upon this analysis, we propose our Font Augmentation method by introducing two sets of fonts, as illustrated in Figure 6. The first set comprises neat and structured fonts, which serve as the standard representations of characters, such as Song (宋体) and Kai (楷体). The second set includes scribble and handwriting fonts, designed to compensate for unconventional writing styles that fall outside the generalization scope of neat fonts, such as Cursive (草书) and Xing (行书), or handwriting fonts Satuo (洒脱) and Qiaomu (乔木).

Particularly, for each training image, we re-render the recognized text using two fonts: one randomly selected neat font and one scribble font. Although this slightly increases training time, it significantly enhances the adaptability of the model towards various handwriting styles.

4.4 Reading Joint Training

As the subtask patterns are settled down, we further move to the their combination, we adopt a vision-language model for the reading joint training to address the challenges at one stop.

As shown in Figure 3, we concatenate instructions S_1 for the target event extraction with S_2 and S_3 for the subtask Event Object Detection and In-Context Text Recognition. The original images I_1 and augmented font images I_2 is inputted after them. After that, these tokens are subsequently merged to create the input x :

$$x = [S_1, S_2, S_3, I_1, I_2] \tag{5}$$

The input x is then feed into the our HVLM to generate the target sequence consist of all the subtasks jointly, among which the output sequence of Event Extraction would be the only part that overall performance would be calculated upon.

5 Experiment

5.1 Dataset and Experiment Setting

We evaluate the performance of our HVLM and other baselines systems on the proposed datasets. For our Vision-Language Model, we use pre-trained QWen-2-VL (Yang et al., 2024) and apply LoRA fine-tuning to the LLM adapter parameters. Hyperparameters are selected via grid search on the validation set, with results averaged over 5 runs. We fine-tune for 30 epochs, saving model parameters for inference. LoRA alpha is set to 128 and rank to 64. Optimization uses Adam (Kingma and Ba, 2015) with a 5e-5 learning rate, batch size 16, and sequence length cutoff of 4096.

5.2 Main Result

In Table 2, we present a comprehensive comparison of our proposed model with cutting edge baselines, include: OCR modules such as PaddleOCR (Du et al., 2021), GOT-OCR2.0 (Wei et al., 2024) and EasyOCR (JaidedAI, 2024) combined with event extraction methods that trained on the textual label of trainset, include

Method	Tri-C F1	Arg-C F1
Baseline (*PaddleOCR+Qwen2.5*)	0.552	0.463
Topline (*Ground Truth+Qwen2.5*)	0.663	0.577
Basic	0.519	0.435
+ Event Object Detection	0.598	0.473
+ Font Augmentation	0.574	0.488
+ In-Context Text Recognition	0.633	0.544
+ *Context Recovery*	0.613	0.526
+ *Typo Revision*	0.534	0.451
+ *Skipping Correction*	0.537	0.462
Ours	**0.648**	**0.553**

Table 3: Contribution of the reading joint training.

classification-based: 1) NPN (Lin et al., 2018), 2) TLNN (Ding et al., 2019), 3) ONEIE (Lin et al., 2020), and generative methods: 1) DEGREE (Hsu et al., 2022), 2) QWen2.5 (Yang et al., 2024), 3) LLaMA3 (AI@Meta, 2024), 4) ChatGLM4 (GLM et al., 2024). We also have end-to-end vision-language models, include LoRA finetuned QWen-2-VL (Yang et al., 2024) and InternLM-XComposer (Dong et al., 2024).

Table 2 indicates that among pipeline methods, generative approaches such as QWen2.5 (Yang et al., 2024) surpass classification-based ones, underscoring the advantage of unified generation architectures that exploit rich label semantics by encoding natural language labels into the target output for extraction. In addition, end-to-end methods outperform most pipeline approaches, highlighting the benefit of avoiding error propagation in complex handwritten event extraction tasks. Moreover, our proposed model exhibits significant improvements over all prior studies ($p < 0.05$), demonstrating the efficacy of our Handwriting Vision-language model when applied on the handwriting images, validating our motivation of proposing Reading Joint Training that simulates the human reading habits of handwriting.

We also have the case studies in Appendix A for a more comprehensive comparison.

5.3 Contribution of Reading Joint Training

After analyzing the overall performance, we further investigate the contribution of RTJ. Particularly, we gradually incorporate various subtasks into our VLM. We use "Basic" in Table 3 to refer the remove of RJT. We add the strongest baseline *PaddleOCR + Qwen* for better illustration. we also have an additional topline *Ground Truth + Qwen* that replace the recognized text with the ground truth for a more comprehensive comparison.

OCR Method	Edit Distance ↓
PaddleOCR	0.083
EasyOCR	0.098
GOT-OCR2.0	0.096
Ours	0.081

Table 4: Text recognition results on different OCR methods. Our method does not have a great lead, indicating our advantage is built upon the extraction step.

Event Extraction Method	Tri-C F1	Arg-C F1
ONEIE	0.452	0.367
DEGREE	0.437	0.362
Qwen2.5-7B	0.544	0.451
LLaMA3-8B	0.443	0.353
Ours	0.648	0.533

Table 5: Results on event extraction methods, all the methods are fed with the context recognized by our method to compare the extraction performance solely.

From Table 3 we can get the distribution: *Topline* > *Ours* > *Baseline* > *Basic*. Without our RJT, the performance of *Baisc* is lower then the *PaddleOCR + Qwen*. On the other hand, *Ours* with RJT can easily surpass the strongest baseline *PaddleOCR + Qwen* and come close to the topline *Ground Truth + Qwen* that takes the golden sentence as input. Besides, all subtasks contribute positively, among which the In-context Text Recognition outperforms the rest, showing superior robustness across diverse noisy inputs.

6 Analysis and Discussion

6.1 Impact of Handwriting Recognition and Event Extraction

As we mentioned in the task challenges (Section 3.3), there are two major steps in our task, the **handwriting recognition** and **event extraction**. We thus investigate that which one is more critical and contributes more to the final performance.

Particularly, we do a pair of comparative experiments: 1) we first compare the text recognition performance of our method and other OCR baselines to check whether our advantage is built upon a supreme recognition ability; 2) we then feed the context recognized by our method into other pure-textual event extraction models to check if our advantage is established in the step of extraction.

From Table 4&5, we observe: 1) Our first hypothesis is not supported—our In-Context Text Recognition performs similarly to other baselines, indicating that our advantage lies in the event extraction step. 2) Consistently, when all meth-

Font	Type	Illustration	CHW-EE			
			Tri-I	Tri-C	Arg-I	Arg-C
Song(宋体)	Neat	甲级联赛	0.671	0.639	0.566	0.538
Kai(楷体)		甲级联赛	0.669	0.635	0.564	0.535
Cursive(草书)	Scribbled	甲级联赛	0.662	0.631	0.538	0.512
Satuo(洒脱)		甲级联赛	0.652	0.628	0.547	0.531
Song(宋体) + Satuo(洒脱)	Mixed	甲级联赛	**0.679**	**0.644**	**0.571**	**0.551**
Song (宋体)+ Cursive(草书)		甲级联赛	0.674	0.643	0.569	0.548

Table 6: Result of different fonts and formations, measured by F1-score.

ods use the text recognized by our approach, our method outperforms the rest. This result shows that our end-to-end model's direct connection between visual and textual modalities effectively reduces error propagation from text recognition.

In general, these results suggest that event extraction is the more critical and challenging step in our task, as it is the key to mitigating error propagation and improving language understanding.

6.2 Impact of Fonts

One of the challenges of our extraction task are the writing styles of the image. We thus investigate if we can improve our HVLM's performance by introducing it various fonts in the trainset. Particularly, we train our HVLM with two sets of fonts for the trainset: 1) Neat fonts that are more considered to be formal and standardized such as Song (宋), Kai (楷). 2) Scribbled fonts are close to the human handwriting such as Cursive (草书).

From Table 6 we can tell that the performances within each group are similar. Among neat fonts, Song outperforms Kai in Arg-I and Arg-C but falls behind in other metrics. A similar trend is seen in scribbled fonts, where Satuo, despite the highest overall performance, lacks consistent superiority.

As shown in Table 6, performances within each group are similar. Between the two groups, neat fonts significantly outperform scribbled ones. Scribbled fonts like Satuo (洒脱) help with specific styles but offer limited generalization. However, combining Song (宋体) with scribbled fonts further improves performance, giving us a conclusion that scribbled fonts help address corner cases within the broad generalization of neat fonts.

6.3 Impact of Handwriting Neatness

We further compare our HVLM with *PaddleOCR + Qwen* to more thoroughly assess its effectiveness on scribbled cases. Concretely, we rank the 10 writers in our test set by overall neatness, ordering them from the neatest to the most heavily scribbled, and we visualize ranks 1, 5, and 10 in Figure 7 for reference. The results indicate that cleaner handwriting consistently yields better performance for both methods. Moreover, as handwriting becomes more scribbled, our model's advantage grows, demonstrating the superiority of HVLM in challenging edge cases where the scribbled or altered handwriting is prone to misunderstanding by traditional OCR-based approaches.

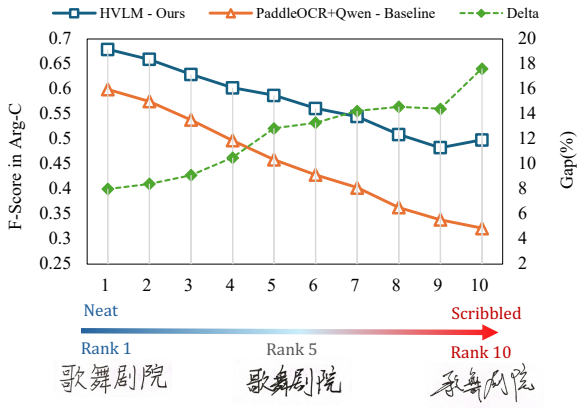


Figure 7: The result on different levels of neatness.

7 Conclusion

In this study, we address the often-overlooked challenge of extracting events directly from handwritten images with a new task: event extraction from handwritten images. We thus introduce **C**hinese **H**and**W**riting **E**vent **E**xtraction (**CHW-EE**) with real human handwriting for benchmarking. With Reading Joint Training, our Handwriting Vision-Language Model significantly outperforms both the pipeline and end-to-end baselines, establishing a strong benchmark for our setting. Our results also show that, for complex real-world tasks, global optimization across the entire process could yield better performance than the traditional pipeline manner of breaking into multiple steps.

Limitations

Our work has two main limitations. First, we do not consider video-form inputs. The current setting only handles static handwritten images, leaving out scenarios such as frame-by-frame whiteboard recordings, classroom or meeting captures with camera motion, and long-duration handwriting videos with zooming, occlusions, and layout drift across frames. These introduce temporal dependencies, motion blur, lighting changes, and cross-frame alignment challenges. A systematic study on aligning visual content with handwriting in videos, exploiting stroke evolution to resolve corrections and typos, and ensuring robustness under realistic compute budgets would clarify the added value of video signals.

Second, we have not extended or evaluated the approach beyond Chinese. Languages such as English with Latin scripts, Arabic with cursive writing, and mixed-language settings may differ in word boundaries, capitalization and punctuation norms that could impact both recognition and event alignment. Broadening to multilingual datasets and assessing performance under cross-language annotation settings would better establish generalization and robustness applicability.

Ethical Statement

For the CHE-EE dataset, we hired 10 annotators to write 100 images each at 2 CNY per image. The work was completed within 2 hours, resulting in an average hourly wage over 100 CNY, well above the local minimum of 19 CNY per hour.

References

AI@Meta. 2024. Llama 3 model card.

Xiaoyi Bao, Jinghang Gu, Zhongqing Wang, Minjie Qiang, and Chu-Ren Huang. 2024. Employing glyphic information for Chinese event extraction with vision-language model. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 1068–1080, Miami, Florida, USA. Association for Computational Linguistics.

Ning Ding, Ziran Li, Zhiyuan Liu, Haitao Zheng, and Zibo Lin. 2019. Event detection with trigger-aware lattice neural network. 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)*, pages 347–356, Hong Kong, China. Association for Computational Linguistics.

Xiaoyi Dong, Pan Zhang, Yuhang Zang, Yuhang Cao, Bin Wang, Linke Ouyang, Xilin Wei, Songyang Zhang, Haodong Duan, Maosong Cao, Wenwei Zhang, Yining Li, Hang Yan, Yang Gao, Xinyue Zhang, Wei Li, Jingwen Li, Kai Chen, Conghui He, and 4 others. 2024. Internlm-xcomposer2: Mastering free-form text-image composition and comprehension in vision-language large model. *Preprint*, arXiv:2401.16420.

Yuning Du, Chenxia Li, Ruoyu Guo, Cheng Cui, Weiwei Liu, Jun Zhou, Bin Lu, Yehua Yang, Qiwen Liu, Xiaoguang Hu, Dianhai Yu, and Yanjun Ma. 2021. Pp-ocrv2: Bag of tricks for ultra lightweight ocr system. *Preprint*, arXiv:2109.03144.

Zhiyuan Fan and Shizhu He. 2023. Efficient data learning for open information extraction with pre-trained language models. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 13056–13063, Singapore. Association for Computational Linguistics.

Team GLM, Aohan Zeng, Bin Xu, Bowen Wang, Chenhui Zhang, Da Yin, Diego Rojas, Guanyu Feng, Hanlin Zhao, Hanyu Lai, Hao Yu, Hongning Wang, Jiadai Sun, Jiajie Zhang, Jiale Cheng, Jiayi Gui, Jie Tang, Jing Zhang, Juanzi Li, and 37 others. 2024. Chatglm: A family of large language models from glm-130b to glm-4 all tools. *Preprint*, arXiv:2406.12793.

I-Hung Hsu, Kuan-Hao Huang, Elizabeth Boschee, Scott Miller, Prem Natarajan, Kai-Wei Chang, and Nanyun Peng. 2022. DEGREE: A data-efficient generation-based event extraction model. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1890–1908, Seattle, United States. Association for Computational Linguistics.

JaidedAI. 2024. Easyocr.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

Jiaqi Li, Chuanyi Zhang, Miaozeng Du, Dehai Min, Yongrui Chen, and Guilin Qi. 2023a. Three stream based multi-level event contrastive learning for text-video event extraction. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 1666–1676, Singapore. Association for Computational Linguistics.

Peifeng Li and Guodong Zhou. 2012. Employing morphological structures and sememes for Chinese event extraction. In *Proceedings of COLING 2012*, pages 1619–1634, Mumbai, India. The COLING 2012 Organizing Committee.

Peifeng Li, Guodong Zhou, Qiaoming Zhu, and Libin Hou. 2012. Employing compositional semantics

and discourse consistency in Chinese event extraction. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 1006–1016, Jeju Island, Korea. Association for Computational Linguistics.

Ruiqi Li, Patrik Haslum, and Leyang Cui. 2023b. EDeR: Towards understanding dependency relations between events. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 14969–14983, Singapore. Association for Computational Linguistics.

Hongyu Lin, Yaojie Lu, Xianpei Han, and Le Sun. 2018. Nugget proposal networks for Chinese event detection. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1565–1574, Melbourne, Australia. Association for Computational Linguistics.

Ying Lin, Heng Ji, Fei Huang, and Lingfei Wu. 2020. A joint neural model for information extraction with global features. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7999–8009, Online. Association for Computational Linguistics.

Jian Liu, Dianbo Sui, Kang Liu, Haoyan Liu, and Zhe Zhao. 2023. Learning with partial annotations for event detection. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 508–523, Toronto, Canada. Association for Computational Linguistics.

Yaojie Lu, Hongyu Lin, Jin Xu, Xianpei Han, Jialong Tang, Annan Li, Le Sun, Meng Liao, and Shaoyi Chen. 2021. Text2Event: Controllable sequence-to-structure generation for end-to-end event extraction. 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)*, pages 2795–2806, Online. Association for Computational Linguistics.

Lei Sha, Jing Liu, Chin-Yew Lin, Sujian Li, Baobao Chang, and Zhifang Sui. 2016. RBPB: Regularization-based pattern balancing method for event extraction. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1224–1234, Berlin, Germany. Association for Computational Linguistics.

Shirong Shen, Guilin Qi, Zhen Li, Sheng Bi, and Lusheng Wang. 2020. Hierarchical Chinese legal event extraction via pedal attention mechanism. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 100–113, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Christopher Walker, Stephanie Strassel, Julie Medero, and Kazuaki Maeda. 2006. ACE 2005 Multilingual Training Corpus. Linguistic Data Consortium.

Haoran Wei, Chenglong Liu, Jinyue Chen, Jia Wang, Lingyu Kong, Yanming Xu, Zheng Ge, Liang Zhao, Jianjian Sun, Yuang Peng, and 1 others. 2024. General ocr theory: Towards ocr-2.0 via a unified end-to-end model. *arXiv preprint arXiv:2409.01704*.

Nuo Xu, Haihua Xie, and Dongyan Zhao. 2020. A novel joint framework for multiple Chinese events extraction. In *Proceedings of the 19th Chinese National Conference on Computational Linguistics*, pages 950–961, Haikou, China. Chinese Information Processing Society of China.

An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, Kai Dang, and 22 others. 2024. Qwen2.5 technical report. *arXiv preprint arXiv:2412.15115*.

MingKun Yang, Yushuo Guan, Minghui Liao, Xin He, Kaigui Bian, Song Bai, Cong Yao, and Xiang Bai. 2019. Symmetry-constrained rectification network for scene text recognition. *Preprint*, arXiv:1908.01957.

Yuqing Yang, Qipeng Guo, Xiangkun Hu, Yue Zhang, Xipeng Qiu, and Zheng Zhang. 2023. An AMR-based link prediction approach for document-level event argument extraction. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 12876–12889, Toronto, Canada. Association for Computational Linguistics.

Deli Yu, Xuan Li, Chengquan Zhang, Tao Liu, Junyu Han, Jingtuo Liu, and Errui Ding. 2020. Towards accurate scene text recognition with semantic reasoning networks. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 12110–12119.

Meishan Zhang, Hao Fei, Bin Wang, Shengqiong Wu, Yixin Cao, Fei Li, and Min Zhang. 2024a. Recognizing everything from all modalities at once: Grounded multimodal universal information extraction. In *Findings of the Association for Computational Linguistics ACL 2024*, pages 14498–14511, Bangkok, Thailand and virtual meeting. Association for Computational Linguistics.

Weiyan Zhang, Wanpeng Lu, Jiacheng Wang, Yating Wang, Lihan Chen, Haiyun Jiang, Jingping Liu, and Tong Ruan. 2024b. Unexpected phenomenon: LLMs' spurious associations in information extraction. In *Findings of the Association for Computational Linguistics ACL 2024*, pages 9176–9190, Bangkok, Thailand and virtual meeting. Association for Computational Linguistics.

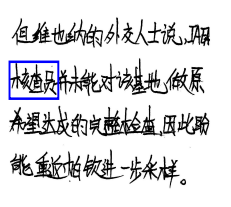
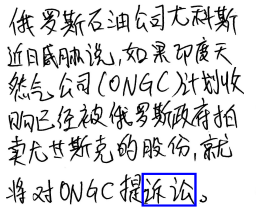
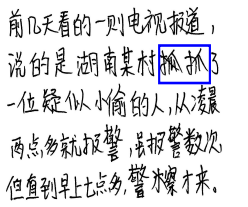
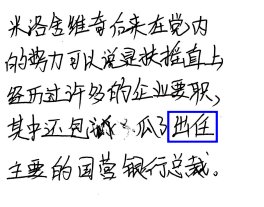
Input Handwriting Image				
Subtask	Argument Identification	Trigger Classification	Trigger Classification	Trigger Identification
PaddleOCR +Qwen	*IA* 日木亥查员 ✗, 帕钦	咸胁 ✗ (Conflict:Attack) ✗	报道 ✗ (Contact:Meet) ✗	酒瓜了出任 ✗ (Personnel:Start-Position)
Ours	核查员, 帕钦	诉讼 (Justice:Sue)	抓 (Justice:Arrest-Jail)	出任 (Personnel:Start-Position)

Table 7: Cases studies, the first two cases are for scribbled writing styles, and the last two are for corrections and typos.

Xinlang Zhang, Zhongqing Wang, and Peifeng Li. 2023. Multimodal chinese event extraction on text and audio. In *2023 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8.

A Case Study

We launch case studies to make a more intuitive comparison between our HVLM and pipeline baselines in Table 7.

We demonstrate that HVLM effectively handles scribbled handwriting, as shown in the first two examples. The pipeline baseline misrecognizes "核" (check) as "木" (wood) and "亥" (noon) in the first example and completely misses the target in the second, while HVLM correctly identifies the elements in both cases. Additionally, we further illustrate generated sentimental images help address common handwriting corrections. In the third example, the correction of "抓" (catch) confuses the pipeline, and in the last example, it wrongly identifies corrections before "出任" (appointment) as the trigger. HVLM avoids these issues, improving extraction accuracy.