# Typeface Generation through Style Descriptions With Generative Models


Pan Wang*
Industrial Design Engineering
Delft University of technology
Delft, Netherlands
p.wang-2@tudelft.nl

Xun Zhang
Laboratory for Artificial Intelligence in Design
The Hong Kong Polytechnic University
Hong Kong, Hong Kong
X.Zhang-16@tudelft.nl

Zhibin Zhou
School of Design
The Hong Kong Polytechnic University
Hong Kong, Hong Kong
zhibin.zhou@polyu.edu.hk

Peter Childs
Imperial College London
London, United Kingdom
p.childs@imperial.ac.uk

Kunpyo Lee
School of Design
The Hong Kong Polytechnic University
Hong Kong, Hong Kong
kunpyo.lee@polyu.edu.hk

Maaike Kleinsmann
Industrial Design Engineering
Delft University of technology
Delft, Netherlands
m.s.kleinsmann@tudelft.nl

Stephen Jia Wang
School of Design
The Hong Kong Polytechnic University
Hong Kong, Hong Kong
stephen.j.wang@polyu.edu.hk

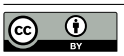


## Abstract

Typeface design plays a vital role in graphic and communication design. Different typefaces are suitable for different contexts and can convey different emotions and messages. Typeface design still relies on skilled designers to create unique styles for specific needs. Recently, generative adversarial networks (GANs) have been applied to typeface generation, but these methods face challenges due to the high annotation requirements of typeface generation datasets, which are difficult to obtain. Furthermore, machine-generated typefaces often fail to meet designers' specific requirements, as dataset annotations limit the diversity of the generated typefaces. In response to these limitations in current typeface generation models, we propose an alternative approach to the task. Instead of relying on dataset-provided annotations to define the typeface style vector, we introduce a transformer-based language model to learn the mapping between a typeface style description and the corresponding style vector. We evaluated the proposed model using both existing and newly created style descriptions. Results indicate that the model can generate high-quality, patent-free typefaces based on the input style descriptions provided by designers. The code is available at: https://github.com/tqxg2018/Description2Typeface



*Corresponding author



*VRCAI '24, December 01–02, 2024, Nanjing, China*



ACM ISBN 979-8-4007-1348-4/24/12
https://doi.org/10.1145/3703619.3706043



## CCS Concepts

• **Computing methodologies** → **Computer vision**; **Information extraction**.

## Keywords

Typeface Design, Typeface generation, Computer vision, Artificial Intelligence, Generative Adversarial Networks

**ACM Reference Format:**
Pan Wang, Xun Zhang, Zhibin Zhou, Peter Childs, Kunpyo Lee, Maaike Kleinsmann, and Stephen Jia Wang. 2024. Typeface Generation through Style Descriptions With Generative Models. In *The 19th ACM SIGGRAPH International Conference on Virtual-Reality Continuum and its Applications in Industry (VRCAI '24), December 01–02, 2024, Nanjing, China.* ACM, New York, NY, USA, 12 pages. https://doi.org/10.1145/3703619.3706043


## 1 Introduction

Designing and creating a new typeface style currently requires the participation of professional typographic designers, as they must combine experience, expertise, and creativity to meet specific design requirements and application scenarios. Fully automating this process remains a challenge in computer vision. This paper introduces a new typeface generation model by incorporating natural language processing into GAN models to address these issues. Our model significantly reduces the need for professional typographers in the typeface design process, enabling everyday users to create patent-free, custom typefaces based on their own descriptions. Additionally, the proposed model can generate typefaces aligned with designers' requirements, offering inspiration and support to professional designers for their creative tasks.

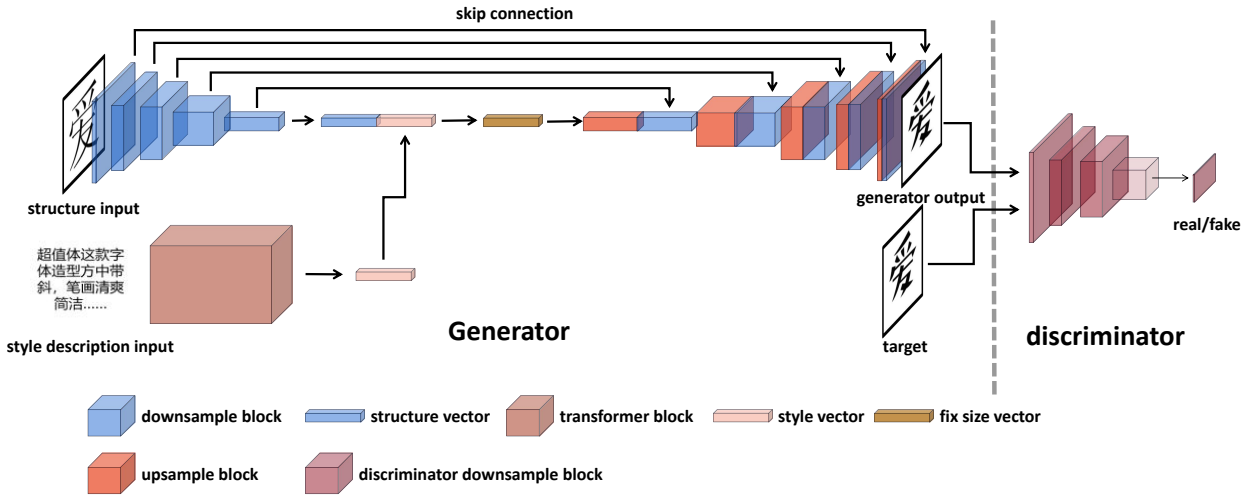



**Figure 1: The overall structure of the proposed Description2Typeface. The model aims to generate a styled Typeface image from an input style description and structure image. The downsample blocks and pre-trained language model encode the structure and style separately, feeding them into upsample blocks to produce the result. The discriminator evaluates the authenticity of the generated image.**

The primary limitations of GAN-based typeface generation are the lack of a well-annotated dataset, the inability to tailor outputs to specific application scenarios, and the absence of research connecting typeface style descriptions with style vectors. To overcome these challenges, we propose **Description2Typeface**, a model that allows users to generate description-specific typeface styles. First, collecting general style descriptions is easier than sourcing finely annotated attributes. Using data from the *Fang Zheng Zi Ku* website, we compiled a new dataset of 400 TTF files with corresponding Chinese style descriptions. Second, we assume that a language model can learn the relationship between typeface styles and their descriptions. Based on this, we introduce a transformer-based, pre-trained language model to map typeface style descriptions to style vectors. Finally, we train a GANs model to transform a given structural typeface image into any style described by the input. During inference, the style encoder encodes a description into a style vector, while the generative model uses a structural typeface image and encodes it into a structure vector. By combining the style vector with the structure vector, the model generates a styled typeface image corresponding to the description. The architecture of our model allows users to enter Chinese descriptions of any length, with the language model ensuring robust generation results. Experimental results on test sets and randomly generated descriptions demonstrate the model's ability to produce high-quality, custom typefaces. Additionally, a user study involving 10 participants in a poster design task indicates that the model effectively generates typeface images with specific styles, assisting both designers and non-designers in their design tasks.

Our main contributions are as follows:

(1) Our model generates high-quality, patent-free typeface images based on a given description, offering substantial benefits to both casual users and professional designers.
(2) This paper introduces a transformer-based, pre-trained language model designed to learn the mapping between typeface style descriptions and style vectors.
(3) This paper introduces a new typeface dataset consisting of 400 styles and their corresponding descriptions, offering a novel pathway for typeface image generation.

## 2 Related Work

### 2.1 Image Synthesis and Style Transfer

Image synthesis generates realistic images using computer graphics and vision methods, particularly through GANs [Goodfellow et al. 2014] and diffusion models [Rombach et al. 2022]. Style transfer, a subtask of image synthesis, requires style input, exemplified by Isola et al.'s [Isola et al. 2017] pix2pix framework using CNNs to extract and apply style features. Zhu et al. [Zhu et al. 2017] introduced CycleGAN for unpaired style transfer, while Karras et al. [Karras et al. 2019] developed a mapping network for controlled face generation. He et al. [He et al. 2019], Liu et al. [Liu et al. 2019a], and Wu et al. [Wu et al. 2019] further explored attribute-controlled generation. Yang et al. [Yang et al. 2022] proposed a GAN-based method for realistic embroidery synthesis, addressing color shift and texture issues.

### 2.2 Language Pre-trained Models

Natural-language processing models are used to understand and analyse sentences. The traditional deep-learning language model usually utilises a recurrent neural network (RNN) as its key component. However, the recently proposed transformer layer has turned out to be more promising in natural-language processing studies. Bidirectional Encoder Representations from Transformers (BERT) [Devlin et al. 2018] is an effective framework that feeds both left

and right contexts into the transformer layers to learn the deep bidirectional representations. Enhanced Representation through kNowledge IntEgration (ERNIE) [Sun et al. 2019] is a new masking strategy for the model training process. Efficiently Learning an Encoder that Classifiers Token Replacements Accurately (ELECTRA) [Clark et al. 2020] uses the Masked Language Model (MLM) as the generator which predicts the masked token. Then, a separate discriminator network predicts the accuracy of the generated token. The Robustly Optimized BERT Pretraining Approach (RoBERTa) [Liu et al. 2019b] fully manifests the power of the original BERT by heavily experimenting with each component of the training process. To further explore the performance and generation of BERT in other languages, Cui et al. [Cui et al. 2020, 2021], Sun et al. [Sun et al. 2021], and Li et al. [Li et al. 2022] tried to adjust the popular pre-trained language model for the Chinese language on tasks such as machine reading comprehension, text classification, etc.

## 2.3 Typeface and Font Generation

Typeface design involves creating a cohesive set of characters with consistent visual features and stylistic elements, forming a unified family, such as Times New Roman or Helvetica. Font design, on the other hand, focuses on the specific implementation of a typeface, including variations in size, weight, and style (e.g., bold, italic) for use in digital or physical formats. Numerous workflows and methods have been proposed for both typeface and font design. Recently, several approaches have framed this as a typeface generation problem, which can be effectively addressed using techniques from computer graphics and computer vision.

*2.3.1 Computer Graphics-based Methods.* Xu et al. [Xu et al. 2009, 2005] proposed a constraint-based reasoning system that uses shape grammar for calligraphy generation. Lin et al. [Lin et al. 2015] use human-labelled attributes to extract components from fonts, and then recombine them to generate new fonts. Lian et al. [Lian et al. 2016] proposed 'FontSL' to reduce human involvement by learning stroke shape and layout separately. But the variety of font styles was affected by the composition rules.

*2.3.2 Deep Learning-based Methods.* Deep learning-based typeface generation is a form of image-to-image translation that transforms a typeface image from one style domain to another, often using conditional generative models. Current approaches are mainly categorized into image-guided and attribute-guided synthesis. Image-guided synthesis uses style features from a reference typeface image to guide generation. For instance, Tian et al. [Tian 2017] proposed a two-stage training approach that first learns general font styles and then fine-tunes on individual font. Jiang et al. [Jiang et al. 2017] introduced DCFont, which generates a full Chinese font set from a small sample of reference images. Attribute-guided synthesis, on the other hand, controls generation with specific typeface attribute values. Wang et al. [Wang et al. 2020] developed Attribute2Font to synthesize font with a given set of attributes. Recently, Yang et al. [Yang et al. 2023] proposed a single-sample font generator based on a diffusion model, while Peong et al. [Peong et al. 2024] and Xiao et al. [Xiao et al. 2024] explored typeface generation with specific styles and logo creation with semantic features, respectively. However, these methods still require users to understand typeface design and attributes. To reduce the need for user expertise, we developed a description-controlled model that generates typefaces based on a single sentence.

# 3 Method Description

## 3.1 Overview

The overall structure of Description2Typeface is shown in Figure 1. The model is designed to generate a typeface image based on an input style description and a structural typeface image in an end-to-end fashion. The generator module receives two inputs: the typeface style description and the structural character image. The structure encoder processes the structural character image to produce a structure vector. Simultaneously, the description encoder, a pre-trained Chinese language model, encodes the style description into a style vector. These vectors are then combined and passed through a series of upsampling layers to generate the styled typeface image. The discriminator module evaluates the authenticity of the generated image. During inference, the model requires only the structural character image and style description to synthesize a new typeface image.

## 3.2 Generator

*3.2.1 Description Encoder Module.* The description encoder module is a Chinese language model pre-trained on popular Chinese natural language processing (NLP) datasets. In this paper, we select several widely used open-source pre-trained models, including ChineseBERT-Large [Sun et al. 2021], MacBERT-Large [Cui et al. 2021], Chinese-RoBERTa-wwm-ext-large [Cui et al. 2021], and Chinese-Pretrain-MRC-MacBERT-Large [Cui et al. 2021]. We evaluate each model's effectiveness in style extraction by comparing the quality of the generated typeface images. The output can be formulated as:

$$V_{\text{text}} = \text{DEM}(d) \tag{1}$$

Where DEM is short for description encoder module. $d$ is the style description. The output vector $V_{\text{text}}$ servers as the typeface style vector.

*3.2.2 Structure Encoder Module.* The structure encoder module consists of a hierarchical structure with multiple downsampling stages. Each stage includes a convolution layer, an instance normalization layer, and an activation function. This process can be described as:

$$V_{\text{image}} = F_d(s) \tag{2}$$

where $F_d$ is the down-sampling function and $s$ is the input structure image. The output vector $V_{\text{image}}$ is the structure vector.

*3.2.3 Image Generation Module.* The image generation module includes a linear projection module and an upsampling module. The linear projection module consists of a linear layer, a dropout layer, and an activation layer. The concatenated vector of $V_{\text{text}}$ and $V_{\text{image}}$ serves as the input to the linear projection module:

$$V_g = F_l([V_{\text{text}}, V_{\text{image}}]) \tag{3}$$

Where $F_l$ is the linear projection function and $V_g$ is the vector combining the style and structure information. The upsampling module is structured hierarchically with multiple upsampling stages. Each

stage consists of a deconvolution layer, an instance normalization layer, and an activation function, except for the final stage, which uses tanh as the activation function. These upsampling stages reconstruct $V_g$ to the original image size. This process can be formulated as:

$$I_g = F_u(V_g) \tag{4}$$

Where $F_u$ is the up-sampling function and $I_g$ is the generated typeface image.

### 3.3 Loss Functions

In the generation step, we define three losses. We use the L1 Norm to measure the pixel-level difference between generation result $I_g$ and target image $I_t$:

$$l_{\text{pixel}} = \|I_g - I_t\| \tag{5}$$

The second loss is contextual loss [Mechrez et al. 2018], an effective measure of similarity between images. Traditional L1 loss assesses similarity at the pixel level, requiring spatial alignment between images for accurate measurement. If images are misaligned, L1 loss is high, even when they are visually similar. Contextual loss addresses this limitation by comparing image features extracted by an image classification model, which contain more information than raw pixels, enabling more accurate similarity measurement. Since image comparison occurs at the feature level, spatial alignment is unnecessary. In the typeface generation task, synthesized typeface images are naturally spatially aligned; however, pixel-level comparison alone cannot effectively capture differences in style, structure, and glyph. Therefore, we supplement L1 loss with contextual loss, formulated as:

$$l_{\text{Contextual}} = \text{Contextual}(I_g, I_t) \tag{6}$$

The final loss is a vanilla generation loss which measures the authentication of the generated typeface image:

$$l_g = -\log p(y_d = 1|I_g) \tag{7}$$

The total loss is formulated as follows:

$$L_G = \lambda_1 l_{\text{pixel}} + \lambda_2 l_{\text{Contextual}} + \lambda_3 l_g \tag{8}$$

where $\lambda_1, \lambda_2, \lambda_3$ are the weights for each loss function.

The discrimination step consists of one discriminator loss which is formulated as:

$$L_D = -\log p(y_d = 1|I_t) - \log p(y_d = 0|I_g) \tag{9}$$

## 4 Experiments

### 4.1 Dataset

We proposed a new typeface dataset containing 400 style pairs, with each pair comprising a TTF file and its corresponding style description[1]. The dataset was collected from the *Fang Zheng Zi Ku* website using a web crawler. Some description sentences in the original raw dataset were blank or unusable, so we refined the data by removing records with empty descriptions and standardizing others into a uniform format.The uniform format for each description sentence translates as: "[Typeface name] style is [typeface style]. Suitable for [applicable scenarios]." This format structure is illustrated in Figure 2.

[The name of the font]字体[The style of the font]。适用于[The scenario this font is suitable for]场景。

**Figure 2: The uniform format for each description sentence. The translation of the sentence is: "*[Typeface name] style is [typeface style]. Suitable for [applicable scenarios]*"**

We use the TTF files to generate Chinese character images. For this paper, we selected the 200 most frequently used Chinese characters as the sample set, creating a dataset of 80,000 images across 400 typeface styles.

### 4.2 Implementation Details

In the experimental stage of the proposed model, the input character image size is $128 \times 128 \times 3$, and the style description input length ranges from a few dozen to several hundred characters. The text tokenizer standardizes the input style description to a fixed length of 512. The output character image size is also $128 \times 128 \times 3$. We use an end-to-end format to train the model. The style description encoder module is selected from the HuggingFace model hub and fine-tuned with a learning rate of 0.00005. The learning rate for the typeface image encoder and decoder modules is set to 0.0002. AdamW is used as the optimizer for the generator model, with a batch size of 16. The input size for the discriminator is $128 \times 128 \times 3$, and its learning rate is set to 0.0002. Model training is conducted on two NVIDIA A100 GPUs.

### 4.3 Evaluation Metrics

We use Inception Score (IS), Fréchet Inception Distance (FID), and pixel-level accuracy (pix-acc) as metrics to evaluate the image generation results. IS assesses the realism and diversity of the generated images, while FID measures the feature vector distance between target images and generated images. To address a limitation of IS and FID, which cannot evaluate image quality at the pixel level, we include pixel-level accuracy (pix-acc) as a third metric.

### 4.4 The Choice of Pre-Trained Language Model

We selected four popular open-source pre-trained language models from the HuggingFace as the style description encoders and evaluated each model's performance individually: ChineseBERT-Large (CL), MacBERT-Large (ML), Chinese-RoBERTa-wwm-ext-large (CRL), and Chinese-Pretrain-MRC-MacBERT-Large (CML). All models are transformer-based and pre-trained on Chinese NLP datasets using distinct training strategies. The transformer-based language model prepends the [CLS] token to the input text, and the output vector corresponding to this token serves as the semantic representation of the input text. In this study, we define the output vector of the [CLS] token as the style description vector, representing the style of the description. Quantitative evaluations of generation results from each pre-trained language model are shown in Table 1, and a visual comparison of the generation results appears in Figure 3.

[1]The dataset is publicly available at https://github.com/tqxg2018/Description2Typeface

**Table 1: Evaluation of IS, FID, and pixel accuracy for generated results, comparing different pre-trained language models with and without fine-tuning.**

| model | IS | FID | pix-acc |
|---|---|---|---|
| CL w/o fine-tune | 2.8971 | 67.7699 | 0.7836 |
| CL w fine-tune | 1.3360 | 310.2761 | 0.5664 |
| ML w/o fine-tune | **2.7159** | **42.7565** | **0.8413** |
| ML w fine-tune | 1.6237 | 296.2735 | 0.5827 |
| CRL w/o fine-tune | 2.7228 | 76.0367 | 0.7628 |
| CRL w fine-tune | 1.2576 | 330.1567 | 0.4915 |
| CML w/o fine-tune | 2.7843 | 48.2853 | 0.8226 |
| CML w fine-tune | 1.5249 | 304.3268 | 0.5336 |


**Figure 3: Comparison of typeface generation results using different pre-trained language models, with and without fine-tuning. The text prompt is *Ben Mo Yue Yi Ti is a creative font with modern characteristics. The font structure is regular but also lively and cute. It is simple, fashionable, elegant and highly functional.***

## 4.5 Attribute Control Typeface Generation

To demonstrate the model's ability to understand style attributes, we conducted an experiment to control the weight of the typeface. Using a baseline style description, we adjusted the keywords that describe the weight, categorizing them into heavy, medium, and thin. The results, shown in Figure 4, show a smooth transition from heavy to thin weights with minimal variation in other attributes.


**Figure 4: Generated results by modifying the key text prompt related to typeface thickness.**

We also tested the model's ability to adapt to different application scenarios. Starting with a baseline description, we adjusted keywords related to the application scenario. The results, shown in Figure 5, demonstrate that the style of the generated typeface changes according to the specified application scenario.


**Figure 5: Generation results by editing the key text prompt related to application scenarios.**

## 4.6 Performance Comparison

Due to the unique nature of this task, we did not find any similar models for typeface generation based on style description in existing work. As a result, we selected an open-source text-to-image generation model, AttnGAN[Xu et al. 2017], for comparison. AttnGAN is an attention-driven, multi-stage model that uses a text vector and random noise as input. We modified AttnGAN's input to include structural information for a more accurate comparison. Both models were trained on the same dataset and evaluated on the same validation set to ensure fairness. As shown in Figure 6, our model demonstrates a superior ability to capture details compared to AttnGAN. Quantitative evaluations of both models are provided in Table 2. Additionally, a detailed assessment of our model's typeface generation capabilities, including comparisons with different style vectors and demonstrations of model diversity, is available in Appendix A.

**Table 2: Comparison of our model against AttnGAN method.**

| model | IS | FID | pix-acc |
|---|---|---|---|
| AttnGAN | 3.6149 | 81.3228 | 0.7961 |
| Description2Typeface | **2.7159** | **42.7565** | **0.8413** |

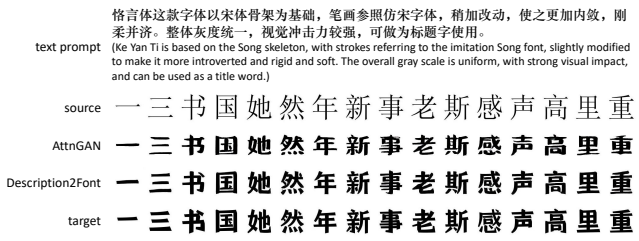



**Figure 6: Comparison of generation results with the AttnGAN method.**

# 5 Case Study

## 5.1 Design Tasks

The proposed typeface generation model was evaluated through a poster design task focused on typeface selection for poster text. Participants were asked to choose or generate typefaces that matched the poster background in both atmosphere and theme. The study included both experienced designers and general participants, emphasizing the suitability of generated typefaces to the poster context while minimizing personal biases. All participants used the same pre-trained typeface generation model. Each poster design task consisted of two steps: participants were given a random background image and Chinese text to add to the poster, then asked to select an appropriate typeface for the text from a selection of existing typefaces or model-generated typefaces.

The experiment consisted of three parts: an introduction session, a trial run session, and the poster design session. The poster design session was divided into two stages: using existing typefaces and using model-generated typefaces. The entire experiment lasted 30 minutes, including a 5-minute introduction, a trial run, and two 10-minute design stages with a 5-minute break between them. During the poster design session, participants used Figma for the design tasks. In the existing typefaces stage, participants selected a suitable typeface from Figma's library of over 300 typefaces, adjusting size and placement as needed. In the model-generated typeface stage, participants provided a description of the desired style to the model to generate corresponding typeface images, made adjustments as needed, and finalized the typeface's size and placement. Each participant completed two posters: one with an existing typeface and one with a model-generated typeface. A more detailed description of the experiment procedure is provided in Appendix B.

Ten students with diverse backgrounds participated in the experiment, ranging in age from 20 to 28, including three males and seven females. All participants were recruited through an announcement on the university bulletin board. To ensure diversity, we included individuals with a mix of backgrounds and work experiences: some participants had formal design education or practical experience in graphic design, such as creating posters for academic or commercial purposes, while others came from various academic and professional fields without specialized design training.

The typeface generation model was trained using over 400 typefaces and their descriptions, collected from the *Fang Zheng Zi Ku* website. The user interface workflow involved creating a description for the typeface style, inputting specific Chinese characters, generating typeface images, and using an automated post-processing system to adjust typography and transparency for ease of use. This model may assist designers and users in finding suitable typefaces efficiently, potentially reducing design costs by lowering typeface copyright fees.

Before each design stage, participants received a brief task introduction to ensure they understood their objectives. The time taken to complete each poster was recorded to assess whether the model could help participants select typefaces more quickly. All posters were collected and shuffled for further analysis.

## 5.2 Data Analysis

To ensure a scientific analysis and validation of the experimental results, we divided the data analysis into three parts: quantitative analysis, participant scoring, and scoring by design experts.

*5.2.1 Quantitative Analysis.* To determine whether participants could find typefaces that met their requirements more quickly using our typeface generation model, we recorded and analyzed the production time for each poster created by all participants. Detailed data are provided in Table 3.

**Table 3: Time taken by participants to create a poster during case study sessions.**

| subject id | time required(seconds) | |
|---|---|---|
| | w/o typeface generation model | with typeface generation model |
| subject 1 | 571 | **467** |
| subject 2 | 540 | **502** |
| subject 3 | 421 | **365** |
| subject 4 | 562 | **419** |
| subject 5 | 487 | **459** |
| subject 6 | 613 | **510** |
| subject 7 | 450 | **427** |
| subject 8 | 579 | **491** |
| subject 9 | 510 | **506** |
| subject 10 | 496 | **430** |
| Average | 522.9 | **457.6** |

*5.2.2 Design Expert Scoring.* The scoring range for each evaluation metric is from 1 to 10, with 10 indicating the highest quality and 1 the lowest. Detailed scoring criteria for each metric are provided in Appendix C.

A total of 20 posters were collected from this user study. Ten of the posters used existing typefaces from Figma, while the other ten featured typefaces generated by the typeface generation model. Two experts in poster design, each with extensive experience and at least three years of design-related coursework, were invited to rate the posters. By comparing the scores assigned by different raters to the same poster, we can assess the level of agreement among judges. The collected posters were randomly shuffled to prevent the raters from knowing which typefaces were created by the model. They were asked to independently rate all posters based on six evaluation metrics.

**Table 4: Percentage of adjacent agreement among raters for each evaluation metric.**

| Metrics | Adjacent agreement |
|---|---|
| Poster typeface aesthetics | 90% |
| Poster typeface design sense | 90% |
| Integrity of poster typeface | 100% |
| The degree of fit between the poster typeface and the background image | 80% |
| character placement | 100% |
| Poster text readability | 100% |

Stemler et al. [Stemler 2004] proposed an adjacent agreement calculation method to better understand consensus among raters on each evaluation metric. According to this method, the two raters are considered to have reached a consensus if the difference in their scores is less than one point. Based on our data analysis, the overall adjacent agreement is 93.33%. Detailed adjacent agreement for each evaluation metric is provided in Table 4.

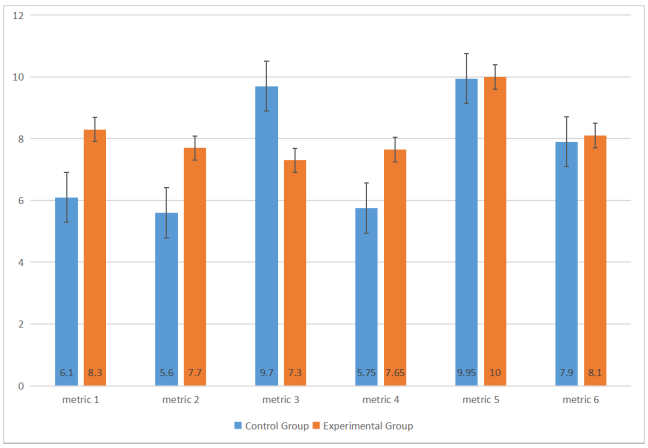



**Figure 7: Average rater scores for each evaluation metric for the control and experimental groups (details of evaluation metrics in Appendix C).**

## 5.3 Results

For data analysis, all ratings from both raters were collected, and the final score for each poster was calculated as the average of the two raters' scores. The Shapiro-Wilk test is a widely used method for assessing whether data follow a normal distribution. Based on the p-values of the rated scores, we determined that the scores did not follow a normal distribution, as the p-value was less than 0.05. The detailed average scores for each poster across all evaluation metrics are shown in Figure 7.

As shown in Figure 7, the average scores for the experimental group posters exceed those of the control group across nearly all evaluation metrics, except for typeface integrity. For quantitative analysis, we recorded the time required to create each poster to assess how quickly participants could find a suitable typeface. The detailed data are presented in Table 3. The experimental results indicate that participants were able to find a suitable typeface more quickly using the typeface generation model, thereby streamlining the poster creation process.

## 5.4 Discussion

Analysis of the ratings collected from design experts shows that the typeface generation model helped participants select appropriate typefaces more quickly, reducing the time needed to create posters. Posters designed with model-generated typefaces outperformed those using existing typefaces across several evaluation metrics, demonstrating improved diversity, design sense, and alignment with designers' needs. To gain further insight into how the model supports designers and reduces their workload, we analyzed each evaluation metric in detail.

First, in terms of time efficiency, posters created with model-generated typefaces took about 60 seconds less on average than those using existing typefaces, as shown in Table 3. This suggests that the model reduces designers' workload by enabling them to find suitable typefaces more quickly. Second, expert ratings indicated that model-generated typefaces scored over two points higher than the control group in aesthetics and design sense. Figure 8 shows examples where control group typefaces appeared overly formal and inconsistent with the relaxed theme of the background, whereas the model-generated typefaces better matched the intended atmosphere, enhancing the poster's overall cohesion.


(a) control group (b) experimental group

**Figure 8: Comparison between control and experimental group. Showing the model-generated typeface better align with the relaxed theme, enhancing poster cohesion compared to the control group's formal typeface.**

In terms of typeface-background fit, the experimental group's typefaces scored approximately two points higher than those in the


(a) control group (b) experimental group

**Figure 9: Comparison between control and experimental group. Showing the model-generated typeface exhibit greater stroke thickness variation and spontaneity, better matching the background and enhancing the poster's message consistency**

control group. Figure 9 shows examples where model-generated typefaces displayed greater variation in stroke thickness and a sense of spontaneity, aligning better with the background image and enhancing the consistency of the poster's message. For character placement, there was no significant difference between the experimental and control groups, as placements were nearly identical. Regarding text readability, both groups performed similarly, indicating that the model-generated typefaces were comparable to existing vector typefaces in readability, as shown in Figure 7. This suggests that the model effectively generates clear and readable typefaces.

Questionnaire responses indicated that 90% of participants preferred model-generated typefaces over existing ones, citing better alignment with their design needs. Additionally, 80% found the user interface easy to use, and 80% felt that the model saved them time in selecting suitable typefaces. These results suggest that the proposed model has a positive impact on designers, particularly in reducing workload and enhancing creativity. Please refer to Appendix D for questionnaire details, and Appendix E for the posters collected from the case study.

In summary, the proposed typeface generation model effectively creates typeface images with specific styles, offering an easy-to-use tool for both designers and non-designers. Compared to existing typefaces, the model significantly reduces workload and enhances aesthetic quality, ultimately supporting creative tasks more efficiently.

## 6 Discussion and Future Work

Our proposed typeface generation model has demonstrated the ability to produce high-quality, customized typefaces based on user-provided descriptions, effectively supporting designers' needs and fostering creativity. However, several aspects warrant further exploration.

First, while our research primarily focuses on the generation of Chinese typefaces, the model's design is adaptable to other languages and typeface styles, provided that suitable datasets are available. This flexibility highlights the model's potential for typeface generation tasks across diverse languages worldwide. Second, our current approach generates typefaces in image format, which limits usability due to structural inconsistencies and the additional processing required to integrate typefaces into design backgrounds. Addressing these limitations by enabling vector-based typeface generation could mitigate these issues and improve the model's practical applications. Third, while the transformer-based pre-trained language model effectively captures style vectors from user descriptions, the quality of generated typefaces heavily depends on the precision of these descriptions, requiring users to have a certain level of prompt engineering skill to achieve optimal results.

Lastly, recent diffusion models [Ho et al. 2020; Nichol and Dhariwal 2021; Sohl-Dickstein et al. 2015] have gained attention for their high-quality outputs and resilience against mode collapse. However, we chose a traditional GAN-based model in this study due to the specific requirements of our task. While diffusion models excel in generating highly detailed and precise images, they often require significantly longer inference times. In contrast, typeface generation tasks prioritize efficiency over extreme detail, making GANs a more suitable choice for faster inference. Nevertheless, we believe diffusion models hold considerable promise for typeface generation, and future work will explore lightweight diffusion models to further enhance performance. Additional future directions include integrating user input with Large Language Models (LLMs) to enable non-expert users to express their ideas more effectively with the support of LLM capabilities. Furthermore, future research will focus on generating vector-based typefaces, expanding to additional languages, enhancing style representation, and incorporating the model into end-user applications such as graphic design software to make creative typography more accessible.

## 7 Conclusion

In this paper, we presented a novel model for generating typeface images based on user-provided style descriptions. By incorporating a transformer-based pre-trained language model to extract style information from text descriptions and leveraging a GAN for typeface image generation, our approach demonstrated significant improvements in the reconstruction detail and completeness of generated typefaces compared to existing methods. The proposed model reduces reliance on professional typographers by enabling both designers and non-experts to create high-quality, patent-free typefaces tailored to specific design needs. Additionally, the model has the potential to inspire professional designers by supporting the creation of innovative typeface styles. Overall, the proposed model shows substantial promise for enhancing creative processes and democratizing typeface design.

## Acknowledgments

The project was funded by the University's Research Centre for Future (Caring) Mobility, the Hong Kong Polytechnic University (P-ID: P0042701); the Collaborative Research Project with World-leading Research Groups (P-ID: P0039528); and the Smart Traffic

Fund (P-ID: P0045764). The research presented in this article was partially funded by grants from the Hong Kong Polytechnic University [Project No. P0042736].

## References


Kevin Clark, Minh-Thang Luong, Quoc V Le, and Christopher D Manning. 2020. Electra: Pre-training text encoders as discriminators rather than generators. *arXiv preprint arXiv:2003.10555* (2020).

Yiming Cui, Wanxiang Che, Ting Liu, Bing Qin, Shijin Wang, and Guoping Hu. 2020. Revisiting pre-trained models for Chinese natural language processing. *arXiv preprint arXiv:2004.13922* (2020).

Yiming Cui, Wanxiang Che, Ting Liu, Bing Qin, and Ziqing Yang. 2021. Pre-training with whole word masking for chinese bert. *IEEE/ACM Transactions on Audio, Speech, and Language Processing* 29 (2021), 3504–3514.

Xun Deng and Liangyan Wang. 2020. The impact of semantic fluency on consumers' aesthetic evaluation in graphic designs with text. *Journal of Contemporary Marketing Science* 3, 3 (2020), 433–446.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805* (2018).

Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2014. Generative adversarial nets. *Advances in neural information processing systems* 27 (2014).

Zhenliang He, Wangmeng Zuo, Meina Kan, Shiguang Shan, and Xilin Chen. 2019. Attgan: Facial attribute editing by only changing what you want. *IEEE transactions on image processing* 28, 11 (2019), 5464–5478.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. 2020. Denoising diffusion probabilistic models. *Advances in neural information processing systems* 33 (2020), 6840–6851.

Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. 2017. Image-to-image translation with conditional adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 1125–1134.

Yue Jiang, Zhouhui Lian, Yingmin Tang, and Jianguo Xiao. 2017. DCFont: an end-to-end deep Chinese font generation system. In *SIGGRAPH Asia 2017 Technical Briefs*. 1–4.

Tero Karras, Samuli Laine, and Timo Aila. 2019. A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 4401–4410.

Hideaki Kawabata and Semir Zeki. 2004. Neural correlates of beauty. *Journal of neurophysiology* 91, 4 (2004), 1699–1705.

Helmut Leder, Benno Belke, Andries Oeberst, and Dorothee Augustin. 2004. A model of aesthetic appreciation and aesthetic judgments. *British journal of psychology* 95, 4 (2004), 489–508.

Linyang Li, Yong Dai, Duyu Tang, Zhangyin Feng, Cong Zhou, Xipeng Qiu, Zenglin Xu, and Shuming Shi. 2022. MarkBERT: Marking Word Boundaries Improves Chinese BERT. *arXiv preprint arXiv:2203.06378* (2022).

Zhouhui Lian, Bo Zhao, and Jianguo Xiao. 2016. Automatic generation of large-scale handwriting fonts via style learning. In *SIGGRAPH Asia 2016 Technical Briefs*. 1–4.

Jeng-Wei Lin, Chian-Ya Hong, Ray-I Chang, Yu-Chun Wang, Shu-Yu Lin, and Jan-Ming Ho. 2015. Complete font generation of Chinese characters in personal handwriting style. In *2015 IEEE 34th International Performance Computing and Communications Conference (IPCCC)*. IEEE, 1–5.

Ming Liu, Yukang Ding, Min Xia, Xiao Liu, Errui Ding, Wangmeng Zuo, and Shilei Wen. 2019a. Stgan: A unified selective transfer network for arbitrary image attribute editing. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 3673–3682.

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

Roey Mechrez, Itamar Talmi, and Lihi Zelnik-Manor. 2018. The contextual loss for image transformation with non-aligned data. In *Proceedings of the European conference on computer vision (ECCV)*. 768–783.

Alexander Quinn Nichol and Prafulla Dhariwal. 2021. Improved denoising diffusion probabilistic models. In *International conference on machine learning*. PMLR, 8162–8171.

KhayTze Peong, Seiichi Uchida, and Daichi Haraguchi. 2024. Typographic Text Generation with Off-the-Shelf Diffusion Model. arXiv:2402.14314 [cs.CV] https://arxiv.org/abs/2402.14314

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. 2022. High-Resolution Image Synthesis with Latent Diffusion Models. arXiv:2112.10752 [cs.CV] https://arxiv.org/abs/2112.10752

Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. 2015. Deep unsupervised learning using nonequilibrium thermodynamics. In *International conference on machine learning*. PMLR, 2256–2265.

Steven E Stemler. 2004. A comparison of consensus, consistency, and measurement approaches to estimating interrater reliability. *Practical Assessment, Research, and Evaluation* 9, 1 (2004), 4.

Yu Sun, Shuohuan Wang, Yukun Li, Shikun Feng, Xuyi Chen, Han Zhang, Xin Tian, Danxiang Zhu, Hao Tian, and Hua Wu. 2019. Ernie: Enhanced representation through knowledge integration. *arXiv preprint arXiv:1904.09223* (2019).

Zijun Sun, Xiaoya Li, Xiaofei Sun, Yuxian Meng, Xiang Ao, Qing He, Fei Wu, and Jiwei Li. 2021. Chinesebert: Chinese pretraining enhanced by glyph and pinyin information. *arXiv preprint arXiv:2106.16038* (2021).

Yuchen Tian. 2017. zi2zi: Master chinese calligraphy with conditional adversarial networks. *Internet] https://github. com/kaonashi-tyc/zi2zi* (2017).

Jan Van Dalen, Henri Gubbels, Charles Engel, Khaya Mfenyana, et al. 2002. Effective poster design. *EDUCATION FOR HEALTH-ABINGDON-CARFAX PUBLISHING LIMITED*- 15, 1 (2002), 79–84.

Yizhi Wang, Yue Gao, and Zhouhui Lian. 2020. Attribute2font: Creating fonts you want from attributes. *ACM Transactions on Graphics (TOG)* 39, 4 (2020), 69–1.

Chuan Wen, Yujie Pan, Jie Chang, Ya Zhang, Siheng Chen, Yanfeng Wang, Mei Han, and Qi Tian. 2021. Handwritten Chinese font generation with collaborative stroke refinement. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*. 3882–3891.

Po-Wei Wu, Yu-Jing Lin, Che-Han Chang, Edward Y Chang, and Shih-Wei Liao. 2019. Relgan: Multi-domain image-to-image translation via relative attributes. In *Proceedings of the IEEE/CVF international conference on computer vision*. 5914–5922.

Shishi Xiao, Liangwei Wang, Xiaojuan Ma, and Wei Zeng. 2024. TypeDance: Creating Semantic Typographic Logos from Image through Personalized Generation. In *Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems* (Honolulu, HI, USA) *(CHI '24)*. Association for Computing Machinery, New York, NY, USA, Article 175, 18 pages. https://doi.org/10.1145/3613904.3642185

Songhua Xu, Tao Jin, Hao Jiang, and Francis CM Lau. 2009. Automatic generation of personal chinese handwriting by capturing the characteristics of personal handwriting. In *Twenty-First IAAI Conference*.

Songhua Xu, Francis CM Lau, William K Cheung, and Yunhe Pan. 2005. Automatic generation of artistic Chinese calligraphy. *IEEE Intelligent Systems* 20, 3 (2005), 32–39.

Tao Xu, Pengchuan Zhang, Qiuyuan Huang, Han Zhang, Zhe Gan, Xiaolei Huang, and Xiaodong He. 2017. AttnGAN: Fine-Grained Text to Image Generation with Attentional Generative Adversarial Networks. arXiv:1711.10485 [cs.CV] https://arxiv.org/abs/1711.10485

Chen Yang, Xinrong Hu, Yangjun Ou, Saishang Zhong, Tao Peng, Lei Zhu, Ping Li, and Bin Sheng. 2022. Unsupervised Embroidery Generation Using Embroidery Channel Attention. In *Proceedings of the 18th ACM SIGGRAPH International Conference on Virtual-Reality Continuum and its Applications in Industry*. 1–8.

Zhenhua Yang, Dezhi Peng, Yuxin Kong, Yuyi Zhang, Cong Yao, and Lianwen Jin. 2023. FontDiffuser: One-Shot Font Generation via Denoising Diffusion with Multi-Scale Content Aggregation and Style Contrastive Learning. arXiv:2312.12142 [cs.CV] https://arxiv.org/abs/2312.12142

Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. 2017. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of the IEEE international conference on computer vision*. 2223–2232.


## A Typeface Generation Capability

We evaluate our model's generation ability on several random descriptions from the validation set. As shown in Figure 10, the generation model successfully captures the style information encoded in the style description and generates the character images with similar structures as the target image.

We also evaluated our model's generation ability by altering the origin of the style vector to validate the effectiveness of the style encoder. As shown in Figure 11, using the style vector generated by the style encoder yields high similarity to the target typeface. In contrast, when a randomly generated style vector is used, the result bears little resemblance to the target typeface. If the style vector is removed entirely from the model structure, the output typeface remains the same as the source typeface without any style modifications.

To verify the diversity of typeface generation in our model, we replaced the style vector from the style encoder with a randomly generated vector. As shown in Figure 12, we display several distinct generation results, with each row representing a set of outputs produced from a unique randomly generated style vector.

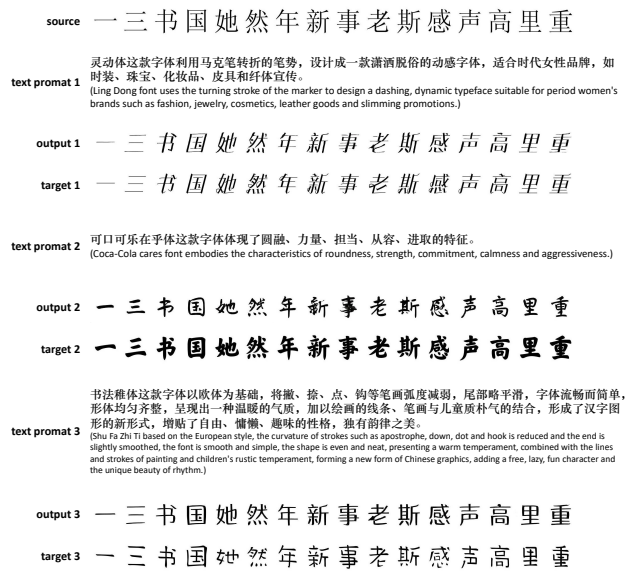



**Figure 10: Typeface generation**

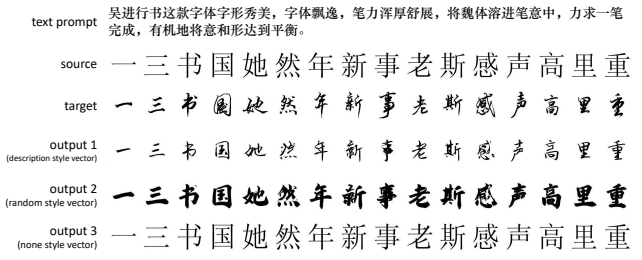



**Figure 11: Results on poster caption typeface design experiments**


**Figure 12: Generation results according to random generated style vectors**

## B Experiment Procedure

The experiment procedure consisted of three parts: an introduction session, a typeface generation model trial session, and a poster design session. The introduction session helped increase participants' understanding of the experiment. The trial session allowed participants to become familiar with using the typeface generation model to generate typefaces based on descriptions. We provided several sets of example descriptions along with corresponding typeface images as references to help participants better understand the model's functionality. In the poster design session, each stage lasted about ten minutes, with a five-minute break between the two stages to reduce participant stress. Figma was chosen as the software for the poster design task.

At the beginning of the experiment, each participant will randomly select an image from a curated library of poster backgrounds. This library contains 50 diverse and representative images sourced from various open-source graphic design repositories and organized by usage scenario. Each background image is paired with a corresponding text, consisting of a short title and descriptive paragraph. To ensure design consistency, each image has been manually matched with the most appropriate text based on the semantic meaning of the text, though some images may share the same description due to differences in the number of available images and texts.

During the first stage, where participants design posters using existing typefaces, they select a suitable typeface from Figma's library of over 300 typefaces. They then adjust the size and placement of the title and description text to match the overall atmosphere and main theme of the poster until they are satisfied.

In the second stage, where participants use the model-generated typeface, they first create a Chinese description that conveys the desired typeface style for the poster. This description is then input into the typeface generation model to produce the desired typeface images. Based on the initial generated images, participants can further refine the Chinese description to better meet their preferences. Finally, they adjust the typeface image size and placement of the title and description text until they are satisfied.

To ensure fairness and consistency in the experiment, the text content itself is fixed and cannot be altered by participants. However, the placement of elements on the poster—such as the title, description, and other graphic elements—can be adjusted according to the participant's design preferences. This approach maintains the integrity of the provided content while allowing creative freedom in layout composition.

## C Evaluation Metrics and Scoring Criteria for Poster Typeface and Overall Aesthetic Quality

To assess both the aesthetics of the posters created by participants and the suitability of the typefaces used, this paper introduces a new set of evaluation metrics for posters and typefaces based on the criteria provided by Van Dalen et al. [Van Dalen et al. 2002]. The proposed metrics are designed to evaluate the typefaces within the posters as well as the posters as a whole. The definitions of each criterion are as follows:

(1) **Poster typeface aesthetics [Kawabata and Zeki 2004; Leder et al. 2004].** Typeface aesthetics refers to the visual appeal of a typeface, including attributes like structural shape, stroke fluency, stroke thickness, and glyph properties. These

features contribute to the overall aesthetic quality of a typeface. As a key visual component, the typeface used in a poster significantly impacts its overall aesthetic appeal, making typeface aesthetics an essential evaluation metric.

(2) **Poster typeface design sense [Deng and Wang 2020].** Typeface design sense refers to the degree to which a typeface demonstrates a unique or creative style beyond simply displaying text. This is often reflected in the distinctive design or modification of structural details within characters, lending them a more innovative appearance. Typefaces with a strong design sense can enhance the design impact of a poster, making its content more visually appealing. Thus, typeface design sense is used as an evaluation metric to compare the design quality of existing and model-generated typefaces.

(3) **Integrity of poster typeface [Jiang et al. 2017; Wen et al. 2021].** Typeface integrity refers to the completeness and clarity of typeface characters, indicating whether any parts appear unclear or incomplete. Existing typefaces in the Figma library are primarily vector-based and generally do not exhibit integrity issues. However, typefaces generated by the model may suffer from some degree of incompleteness due to the direct use of image generation models. Such defects in character integrity can negatively impact the structure and design of the typeface, potentially affecting the intended message conveyed by the poster.

(4) **Fit between the poster typeface and background image [Van Dalen et al. 2002].** The degree of fit between a poster's typeface and its background image assesses whether the chosen typeface aligns with the mood or message of the background image. Various elements like content, color schemes, and themes in the background image convey specific emotions, and selecting a typeface that complements these elements can create a unified emotional tone. A cohesive typeface-background pairing enhances the viewer's experience by making the poster's message and emotions more accessible. Conversely, a mismatched pairing may hinder the viewer's interpretation and reduce the poster's impact.

(5) **Character placement [Van Dalen et al. 2002].** Character placement refers to the positioning of text on a poster's background image. Since the focal point of each poster background varies, optimal placement ensures that text does not obstruct key visual elements, allowing for synchronized text and image information. Thoughtful character placement enriches the poster's overall composition, while incorrect placement may obscure essential elements, diminishing both message clarity and visual appeal.

(6) **Poster text readability [Van Dalen et al. 2002; Wen et al. 2021].** Poster text readability refers to how easily viewers can read text embedded in the poster's background. Well-designed posters typically adjust typeface characteristics, such as size and line thickness, to enhance readability. Text is crucial for conveying a poster's message; clear, readable text supports effective communication, while difficult-to-read text can obstruct information transfer, impacting the viewer's understanding.

For each evaluation metrics mentioned above, the scoring criteria for each evalution metrics are as follows:

(1) Poster typeface aesthetics
   (a) **Score 10 to 9:** Poster typeface have strong aesthetics.
   (b) **Score 8 to 6:** Poster typeface have certain aesthetics.
   (c) **Score 5 to 3:** Poster typeface have normal aesthetics.
   (d) **Score 2 to 1:** Poster typeface have almost no aesthetics, which are commonly used default typeface.

(2) Poster typeface design sense
   (a) **Score 10 to 9:** Poster typeface has strong design sense.
   (b) **Score 8 to 6:** Poster typeface has certain design sense.
   (c) **Score 5 to 3:** Poster typeface has normal design sense.
   (d) **Score 2 to 1:** Poster typeface has almost no design sense.

(3) Integrity of poster typeface
   (a) **Score 10 to 9:** Poster typeface has strong integrity, all positions of characters have good integrity.
   (b) **Score 8 to 6:** Poster typeface has a certain degree of integrity, only a small typeface of the typeface character connection is not smooth.
   (c) **Score 5 to 3:** Poster typeface has normal integrity, some typeface character parts are missing at the junction.
   (d) **Score 2 to 1:** Poster typeface integrity is poor, most of the typeface characters have a lot of jerky and missing detail problems.

(4) The degree of fit between the poster typeface and the background image
   (a) **Score 10 to 9:** The poster typeface fits very well with the background image, which can fully show the emotion and information of the background image, and has a sense of unity.
   (b) **Score 8 to 6:** The poster typeface fits well with the background image, which can partially show the emotion and information of the background image, but the sense of unity is a bit poor.
   (c) **Score 5 to 3:** The poster typeface hardly matches the background image, and only shows a little emotion and information of the background image, which makes the sense of unity a bit poor.
   (d) **Score 2 to 1:** The poster typeface does not match the background image, does not show the emotion and information of the background image, and has no sense of unity.

(5) Character placement
   (a) **Score 10 to 9:** The placement of the characters is easy to read and there is no excessive masking of the background image, which does not disturb the user's perception.
   (b) **Score 8 to 6:** The placement of characters can be improved to a certain extent, and there is a certain mask for the background image, which slightly affects the user's perception.
   (c) **Score 5 to 3:** There are some unreasonable situations in the placement of characters, and there is a mask on the background image, which affects the user's perception.
   (d) **Score 2 to 1:** The position of the characters is very unreasonable, and there is a large area of masking on the important positions of the background image, which greatly affects the user's perception.

(6) Poster text readability

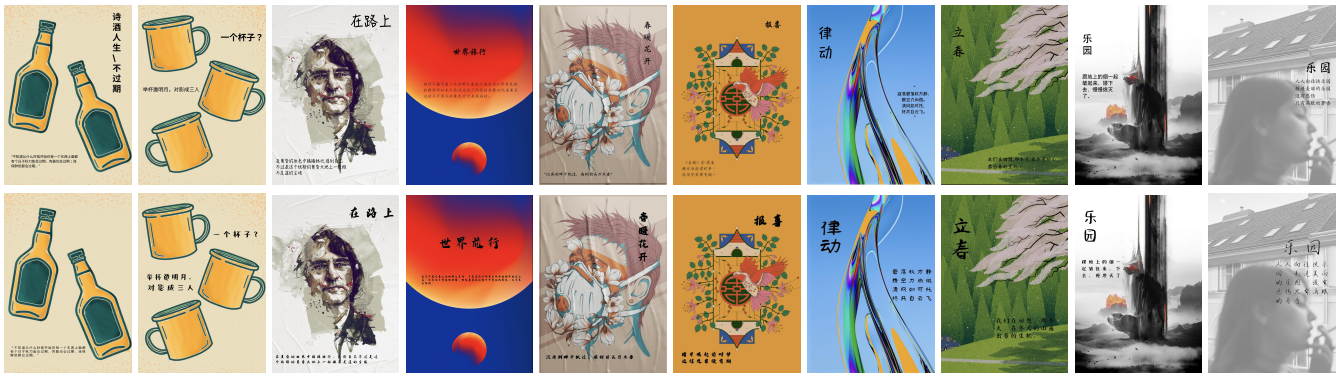



**Figure 13: The posters collected from the case study. The top image shows the posters from the control group, while the bottom image shows the posters from the experimental group.**

(a) **Score 10 to 9:** The text in the poster is very readable.
(b) **Score 8 to 6:** Most of the text in the poster is easy to read, but it does not affect the reading feeling.
(c) **Score 5 to 3:** There are some unreadable parts of the text in the poster, which have a certain impact on the reading experience.
(d) **Score 2 to 1:** The text in the poster is not easy to read at all, which has a great impact on the reading experience.

## D Questionnaire Details

A questionnaire survey for participants is primarily used to assess their experiences with the typeface generation model and to determine whether the model makes finding a suitable typeface easier. After completing the trials, participants share their thoughts on the model's user experience and typeface generation ability based on several evaluation metrics. The details of each evaluation metric are as follows:

(1) Participants are asked to choose which typeface, from either the existing Figma library or the generated typefaces, best meets their requirements. This evaluates the model's ability to generate typefaces that align with human style descriptions. This metric also helps determine whether the generated typeface images are superior to the existing Figma typefaces.
(2) Participants are asked to evaluate the convenience of the model's usage, providing insights into its usability.
(3) Participants indicate whether the typeface generation model could assist them in designing better posters. This metric assesses the model's potential to support designers in their creative tasks.

## E Posters Collected From the Case Study

All posters collected from the case study are shown in Figure 13, showcasing the model's practical application and potential.

## F Example with typeface integrity problem

Issues with typeface integrity were observed in the model-generated typefaces. While existing typefaces, typically vector-based, ensured complete character integrity, model-generated typefaces occasionally exhibited missing or incomplete strokes, resulting in a lower integrity score (7.3). Figure 14 provides examples of incomplete characters.


(a) Poster A (b) Poster B

**Figure 14: Posters with low typeface integrity scores.**