User Demand Prediction and Solution Generation in Textile Product Development and Design Based on Machine Learning

Na Zhu
School of Design and Art, Harbin University of Commerce, Harbin 150001, Heilongjiang, China
Corresponding author: Na Zhu (e-mail: 102554@hrbcu.edu.cn)

ABSTRACT Accurate translation of heterogeneous user requirements into manufacturable design solutions remains a challenging task in intelligent product engineering systems. This study proposes a process-constrained multimodal conditional variational autoencoder (PC-MCVAE) framework for demand representation, solution generation, and engineering feasibility optimization. The proposed architecture integrates multimodal information from textual descriptions and visual inputs through contrastive representation learning to construct a unified latent demand space. To incorporate domain knowledge into the generation process, a process knowledge graph is transformed into differentiable constraint functions that characterize feasible manufacturing regions and are embedded directly into the optimization objective. A conditional variational decoder is then employed to generate hybrid design solutions consisting of discrete structural representations and continuous engineering parameters. Furthermore, an end-to-end training strategy is developed to jointly optimize semantic consistency and process compliance. Experimental results demonstrate that the proposed framework achieves a semantic alignment score of 0.782, a process compliance rate of 91.4%, and a Top-3 user preference prediction accuracy of 83.7%. The model exhibits strong robustness across different product categories and application scenarios while maintaining high generation quality and manufacturing feasibility. The proposed framework provides an effective methodology for multimodal information fusion, knowledge-guided generative modeling, and intelligent decision support in engineering-oriented design systems.

INDEX TERMS multimodal information fusion, conditional variational autoencoder, knowledge graph embedding, intelligent design optimization

I. INTRODUCTION

THE evolution of the textile product development process is experiencing a major change as it shifts away from largescale standardisation towards more personalised solutions that are based upon consumer needs. Today's consumers are becoming more sophisticated in their requests for product features such as look, touch, function, and emotional attachment, which is driving designers to increasingly use data and intelligent tools to aid them in creating these products. Therefore, the ability to convert detailed consumer needs into manufacturable designs while being efficient and accurate has become a major obstacle in the development of intelligent textiles, increasing the competitive advantage of products, and decreasing time-to-market for designers [1,2]. The dual issues of the "semantic gap" and the "manufacturing gap" that exist within the current methodology present a challenge. The "semantic gap" is associated with the user's needs for product that have multi-attributes and can be expressed in open-format text. Traditional statistical models that utilize data collected from product/brand preference surveys, customer purchase histories, etc., only capture limited information about a user's preferences. Therefore, it can be difficult to conduct analysis on user-generated content such as product descriptions or requirements, which may contain user-expressions for example, "lightweight yet tight-fitting" when assessing a new textile for use in apparel. Conversely, the "manufacturing gap" relates to our current solutions delivery mechanisms, which rely extensively on rule-based and template matching approaches, while lacking the ability to model the relationship between material attributes (e.g. yarn twist and density) and the various elements of each industrial process required to produce an item of apparel (e.g. jacquard needle count, dyeing capacity). This lack of inter-model relations leads to generation of solutions that fall outside the ability to produce and practically deliver them to customers as requested. The two associated gaps create a large void with regard to the information necessary to satisfy user needs and, by extension, the functional capability of intelligent design system solutions to meet practical production requirements. Numerous studies have explored the introduction of deep learning into research aimed at reducing this gap. Two main approaches exist in current literature. The first approach employs CNNs for demand analysis and visual-language pre-trained models such as CLIP for cross-modal matching; however, these models' embedded spaces do not include domain knowledge, or any domain-specific parameters, which contributes to a disconnect between the semantic representation and downstream model parameterization [3-5]. Similarly, various types of methodology exist that generate architectural form factors or color specifications using GANs, or VAEs on the production side; however, the generated forms are created without clearly identified engineering processes, meaning that the probability of receiving parameters that will not allow for future utilization or cannot be manufactured - is a large factor in this misunderstanding [6,7]. Although both forms of immediate correction after the post-processing step have attempted to utilize process rules for passive correction, the "passive correction" technique, similar to that of post-processing, does not permit a model to actively converge to the area of permissible production parameters while still being in the generation stage [8-10], thus reducing the difference between these two methods, which ultimately has become increasingly evident as a significant potential bottleneck. Therefore, the opportunity to combine a single framework that can simultaneously include the fine-grained semantic representation of a product while supporting the continuous schematizing processes leading to the manufacture of that same product - has become increasingly apparent to be an enormous challenge. This research paper describes a Process-Constrained Multimodal Conditional Variational Autoencoder (PC-MCVAE) that features process constraint embeddings. Specifically, the PC-MCVAE was created in order to create a framework for mapping user requirements (fine-grained) to manufacturable design solutions in a way that reduces or eliminates any end-to-end differentiability issues with this mapping. The process for creating this model started with a convergence step where the user's textual requirements were aligned with the user's image-based design requirements through contrastive learning. This created one unified embedding for both requirement types in the same latent space. Next, this combined embedding served as a condition to influence a variational decoder to create an output of a hybridized design solution that contained elements (both continuous and discrete) of the final design. The primary innovative aspect of this methodology is the conversion of the textile process knowledge graph's formalized feasible domain boundary into a differentiable penalty. This penalty can then be embedded into the objective function used in the variational inference process. The embedding of the textile process knowledge graph's feasible domain boundaries into the variational inference objective function will allow the generative process to skip over the creation of design solutions that fall into those nonviable, infeasible regions of the textile process knowledge graph while still ensuring maximum semantic alignment between user requirements and generative designs. The proposed PC-MCVAE will allow for the differentiable hybridization or fusion of process knowledge within the generative model establishing a structured and unique differentiable mapping for open-domain user semantics through to the closed manufacturing parameter domain.

II. MULTIMODAL CONDITIONAL VARIATIONAL AUTOENCODER WITH EMBEDDED PROCESS CONSTRAINTS

A. OVERALL FRAMEWORK OVERVIEW

The PC-MCVAE is an end-to-end generative modelling framework that addresses the mapping challenge between fine-grained user requirements and manufacturable solutions for textile product development and design. Three modules are incorporated into the system. The first module performs multi-modal user requirements semantic embedding. The second module conducts differentiable modelling of textile process constraints. The third module handles conditional generation and embedding of process constraints. The multi-modal user requirements embedding module aligns user textual descriptions with product photography using contrastive learning methods. This alignment produces a blended representation of user requirements in a common latent space. The process constraint modelling transforms the textile domain knowledge graph into soft constraint functions via differentiable modelling techniques. The conditional generation module employs the user requirement representation to direct a variational decoder. The decoder produces a hybrid design solution consisting of both discrete pattern encodings and continuous fabric parameters. Process constraints are embedded into the training target via reconstructing the Evidence Lower Bound (ELBO). These three modules work together to achieve a structured and differentiable mapping from open-domain user semantics to closed-domain manufacturing parameters, while ensuring semantic matching and manufacturing feasibility of the solutions.

B. CORE MODULE 1: MULTIMODAL USER REQUIREMENT SEMANTIC EMBEDDING

1) Text and Image Encoder

User feedback data originates from product reviews on e-commerce platforms. Each record contains a free text description and a high-resolution product image of 1024×1024 pixels. The text is lowercase, special symbols and stop words are removed, and it is truncated to a maximum of 64 words. The image is cropped at the center and normalized to the [0,1] interval before being input into the visual encoding module. The text encoder adopts the BERT-base architecture [11,12], with a word embedding dimension of 768, containing 12 Transformer layers and 12 attention heads. All parameters are updated end-to-end during training. The visual encoder is based on the ResNet-50 [13-15] backbone network, removing the original classification layer and retaining the global average pooling output to generate a 2048-dimensional image feature vector. Let the text and image encoding outputs be h_t and h_v, respectively. To achieve cross-modal semantic alignment, two learnable linear projection matrices W_t and W_v are introduced to map heterogeneous features to a shared 512-dimensional latent space, resulting in $z_t = W_t h_t$ and $z_v = W_v h_v$.

2) Alignment of Shared Latent Space Based on Contrastive Learning

In formula (1), the text-to-image contrast loss is defined, and positive and negative pairs are constructed using in-batch samples: positive samples are text-image pairs from the same user, and negative samples are the remaining images in the same batch. Cosine similarity [16,17] is used for similarity, the temperature coefficient τ is set to 0.07, and the loss function is:

$$L_{t2v} = -\log \frac{\exp(\cos(z_t, z_v)/\tau)}{\sum_{k=1}^{N} \exp(\cos(z_t, z_{v,k})/\tau)} \tag{1}$$

In formula (2), the symmetrical image-to-text contrast loss is defined as follows:

$$L_{v2t} = -\log \frac{\exp(\cos(z_v, z_t)/\tau)}{\sum_{k=1}^{N} \exp(\cos(z_v, z_{t,k})/\tau)} \tag{2}$$

The total contrastive loss is the average of the two features, i.e., $L_{contrast} = (L_{t2v} + L_{v2t})/2$. This loss drives semantically similar cross-modal samples to move closer to each other in the latent space, while semantically unrelated samples move further apart. The final demand vector is taken as the arithmetic mean of the projected features: $c = (z_t + z_v)/2$, which serves as the conditional input to the generation module. This vector is formed through data-native semantic alignment without any manual attribute label supervision, covering fine-grained preference information such as color, texture, and function. The projection layer is initialized with Xavier, the optimizer is AdamW, the learning rate is set to 2×10^{-5}, and the weight decay is 0.01. The text encoder, visual encoder, and projection matrix are optimized during combined end-to-end training. An ablation study found that the 512 latent space was optimal with respect to computational efficiency versus expressive capability. It translates unstructured multimodal feedback into a structured and differentiable demand representation generating conditionally accurate downstream generation.

C. CORE MODULE TWO: DIFFERENTIABLE MODELING OF TEXTILE PROCESS CONSTRAINTS

1) Construction of Textile Process Knowledge Graph

A graph for textile processing knowledge was created to categorize the ability to manufacture goods [18-20]. The graph has three kinds of items; fiber types, weaving conditions, and finishing attributes. The connections or edges between the areas of the graph are based on limitations to the methods used. The graph is built from analyses of existent structured domain documents, (standards, manuals, databases) and the use of a strategy called "rule-matching" to extract entities and relationships provides confirmation of the scientific validity and completeness of the knowledge represented. The result of this process is that the graph will form an attribute graph with direction reactivity that provides an overall roadmap to differentiate each of the manufacturing methods.

The subgraph of the process knowledge graph shown in Figure 1 focuses on the core entity "COTTON FIBER" and presents some key process parameters associated with cotton fibre and how these parameters interconnect with each other throughout the different phases of the textile production chain. For example, the node for "COTTON FIBER" has an arrow with the label "max_ends_per_inch" pointing towards a green circular node labelled "WARP DENSITY", indicating that the maximum number of warp ends allowed for cotton fibre is $\leq$ 220 ends/inch. Likewise, the node for "WEFT DENSITY" is connected to the "COTTON FIBER" node with an arrow labelled "max_picks_per_inch", indicating that the maximum number of weft picks allowed is $\leq$ 180 ends/inch. An edge labelled "suitable_for" connects "COTTON FIBER" to the orange diamond node labelled "PLAIN WEAVE STRUCTURE", indicating that cotton fibres can be used in the plain weave structure as a suitable method of weaving cotton fibres. The plain weave structure is connected by an edge labelled "requires_control" to the orange diamond node labelled "PRE-SHRINKAGE RATE", demonstrating that the plain weave method affects the finishing process by requiring that the pre-shrinkage be controlled within the range of 3%. Finally, the "PRE-SHRINKAGE RATE" node is connected to two "DYEING TEMPERATURE" nodes through the "influences_process" edge, indicating that the pre-shrinking treatment will affect the subsequent dyeing process, and together they stipulate that the dyeing temperature must be controlled within the range of 60–90°C, demonstrating the interconnected and progressive process constraint chain from raw material selection and weaving parameter setting to finishing process conditions.

2) Mathematical Formalization and Differentiability Transformation of Constraint Rules

For the continuous process parameter vector $p \in \mathbb{R}^d$ (dimension $d = 7$) output by the generative model, its engineering feasible region F is jointly defined by multiple inequality rules contained in the graph (such as "the upper limit of the warp density of cotton fibers is 220 threads/inch"). The boundary of this feasible region is composed of a series of piecewise linear hyperplanes. In order to enable the constraints to be embedded in the end-to-end training process, the hard feasible region is transformed into a soft constraint function that is differentiable everywhere. As defined by formula (3):

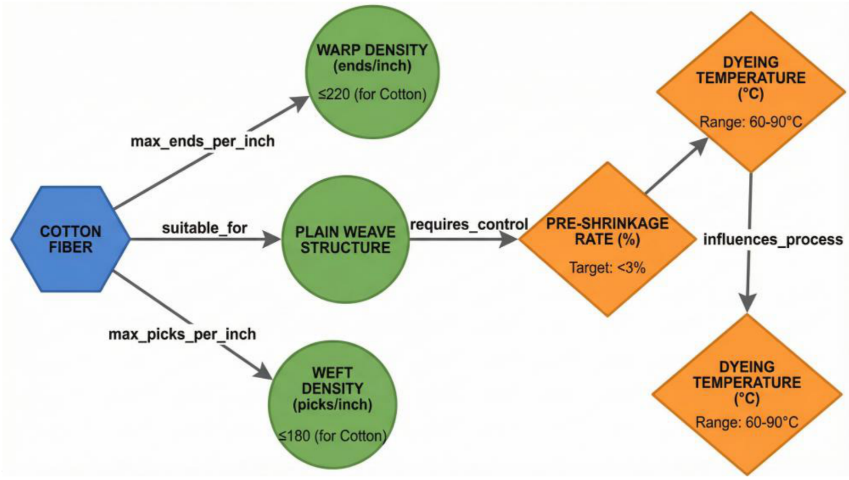

FIGURE 1. Process knowledge graph subgraph used for differentiable constraint modeling

$$\phi(p) = \prod_{i=1}^{n} \sigma\big(\alpha \cdot \mathrm{dist}(p, \mathrm{boundary}_i)\big) \tag{3}$$

In formula (3), $\mathrm{dist}(p, \mathrm{boundary}_i)$ is the signed distance from parameter vector p to the i-th linear constraint boundary (positive when the constraint is satisfied, negative when it is violated), α is the hyperparameter controlling the penalty steepness, and σ is the Sigmoid function. The output value of this function ϕ(p) approaches 1 in the feasible region and rapidly decays to 0 in the infeasible region. Its non-zero gradient near the boundary can effectively backpropagate, providing a clear optimization direction for the generative model. For discrete parameters (such as organizational structure), their legal value set is determined by the "applicability" relationship in the graph and is applied to the Gumbel-Softmax output layer in the form of a mask [21,22] to ensure the process compliance of the generated results. This scheme transforms unstructured domain knowledge into a mathematical expression that can be embedded in the objective function, providing explicit manufacturing feasibility guidance for the generative model.

D. CORE MODULE THREE: CONDITIONAL SCHEME GENERATION AND CONSTRAINT EMBEDDING

1) Demand-Based Variational Decoder

PC-MCVAE models the generation process of textile design scheme x using a user demand vector c as conditional input. Here, x_d represents the discrete pattern encoding, corresponding to a predefined set of pattern primitives, and employs a Gumbel-Softmax mechanism to achieve differentiable discrete sampling. x_c represents continuous fabric physical parameters, including warp density (threads/inch), weft density (threads/inch), weight per unit area (g/m^2), warp count (Ne), weft count (Ne), reed count (teeth/inch), and pre-shrinkage rate (%). A latent variable z is introduced, with its prior distribution set as a standard normal distribution $p(z) = N(0, I)$. The decoder f_θ consists of a four-layer fully connected network, each layer being 1024-dimensional, with the activation function GELU, and the input being the concatenated vector $[c; z]$. The decoder output is processed in two paths: the discrete branch generates x_d through a Gumbel-Softmax layer (with a fixed temperature coefficient of 0.5), and the continuous branch directly outputs x_c through a linear layer. This generation mapping is formally represented as:

$$p_\theta(x \mid c, z) = p_\theta(x_d \mid c, z) \cdot p_\theta(x_c \mid c, z) \tag{4}$$

θ is the learnable parameter of the decoder. The encoder q_ϕ uses four layers of the same depth (Multilayer Perceptron, MLP) [23,24], and outputs the mean μ and log standard deviation $\log \sigma$ of the Gaussian posterior distribution for reparameterization sampling. All network layers are initialized with He, and the hidden layers are followed by LayerNorm, with no Dropout. The overall architecture of the model is shown in Figure 2, which presents the data flow path from multimodal encoding and conditional decoding to process constraint feedback, providing a structural basis for embedding manufacturing feasibility in the subsequent objective function.

2) ELBO Reconfiguration Target with Embedded Process Constraints

The standard conditional variational autoencoder (PC-MCVAE) framework aims to learn a generation distribution conditioned on user demand c. As defined in Section 2.4.1, PC-MCVAE establishes a conditional generation process using latent variables z, where the decoder defines the conditional likelihood distribution $p_\theta(x \mid c, z)$. To derive the Evidence Lower Bound (ELBO), it considers the variational inference objective under the complete data distribution. By integrating over the latent variables z, it obtains the marginal conditional distribution $p_\theta(x \mid c) = \int p_\theta(x \mid c, z) p(z)\, dz$, whose canonical expression for the Evidence Lower Bound is:

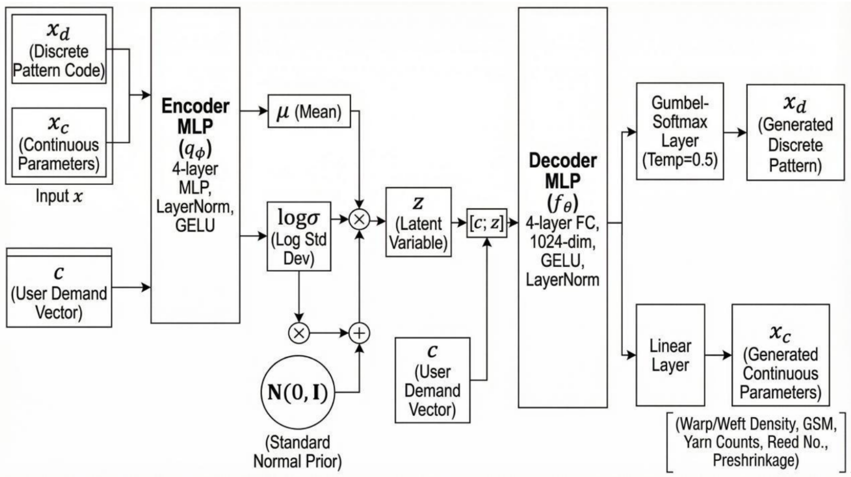

FIGURE 2. Overall architecture diagram of PC-MCVAE

$$
\begin{aligned}
L_{EB} = E_{q_\phi(z|x,c)}\left[\log p_\theta(x_d \mid c, z) + \log p_\theta(x_c \mid c, z)\right] \\
- D_{KL}\left(q_\phi(z \mid x, c) | \, p(z)\right)
\end{aligned}
\tag{5}
$$

The encoder defines an approximate posterior $q_\phi(z \mid x, c)$, with the prior set as a standard normal distribution $p(z) = N(0, I)$, and the decoder defines a likelihood model $p_\theta(x \mid c, z)$. This objective only optimizes data reconstruction capability and the regularity of latent variable distribution, without imposing constraints on the manufacturing feasibility of the generated scheme. To explicitly embed process knowledge, the differentiable constraint function defined in Section 2.3.2 is used, and $\phi(x_c)$ is introduced as a feasibility likelihood term into the objective function, reconstructing it into formula (6):

$$
L_P = L_{EB} + \lambda \cdot E_{q_\phi(z|x,c)}\left[\log \phi(x_c)\right] \tag{6}
$$

In formula (6), λ is the constraint weight, set to 1.0. This penalty term significantly reduces the objective function value when x_c violates the process rules, and its gradient propagates back to the decoder parameters through the chain rule, driving the generation distribution to shrink towards the feasible region. Since $\phi(x_c)$ is differentiable everywhere and explicitly associated with the continuous parameter x_c, this design achieves embedded optimization of manufacturing constraints during the generation process, rather than posterior filtering. The final optimization objective is to maximize LP, which is equivalent to minimizing the sum of the lower bound of negative evidence and the feasibility penalty. This reconstruction enables the model to structurally avoid unmanufacturable parameter combinations while maintaining semantic fidelity. To intuitively demonstrate the behavioral characteristics of the differentiable constraint function $\phi(p)$, Figure 3 shows its response curve on a one-dimensional process parameter (such as the warp density of cotton fibers), presenting how the function achieves a smooth transition from the feasible region to the infeasible region near the hard constraint boundary. Its non-zero gradient provides clear directional guidance for model optimization.

In Figure 3, the response curve to the precision x_{warp} factor is shown for the differentiable constraint function $\phi(x_c)$, visually representing its expressed behavior. The solid blue line is an indication of the function relationship between feasibility score $\phi(x_{warp})$ and precision, whereas the hard constraint boundary $x_{warp} \leq 220$ is indicated by the dashed black line. When $x_{warp} \leq 215$, $\phi(x_{warp})$ remains constant at 1.0, thus suggesting that this parameter satisfies all requirements of the manufacturing process, at least based on this function. The transition region (215-225) displays a smooth diminishing curve for the function from a value of 1.0 for $x_{warp} = 215$ down to 0.0 at a value of 225 for x_{warp}, defining the transition band of ≈ 10 ends/inch, which permits smooth changing of the curve and prevents the abrupt change problems traditionally exhibited with hard constraints while still maintaining differentiability. However, for $x_{warp} > 225$, $\phi(x_{warp}) \approx 0.0$ for x_{warp} greater than or equal to 225 indicates serious violations of manufacturing specifications. Within the transition band (215–225 ends/inch), the curve maintains a non-zero derivative, which provides the necessary backpropagation signals during the optimization process so that the decoder outputs will eventually converge toward the feasible region. Additionally, the transition band design width provides an optimal balance between a narrowly enforced set of manufacturing specifications and the feasibility of conducting optimization thereby enabling effective utilization of the 220 constraint boundary without suffering from the training difficulties normally associated with hard constraints. In addition, it correlates with the engineering knowledge of Textile Manufacturing, where as the parameter departs from the nominal values; the feasibilities of testing would be reduced gradually as well. The Gradient retained from the near boundary positions can allow for the method to avoid these infeasible parameter spaces throughout the entire training process, rather than filtering for these after the training process. This "embedded

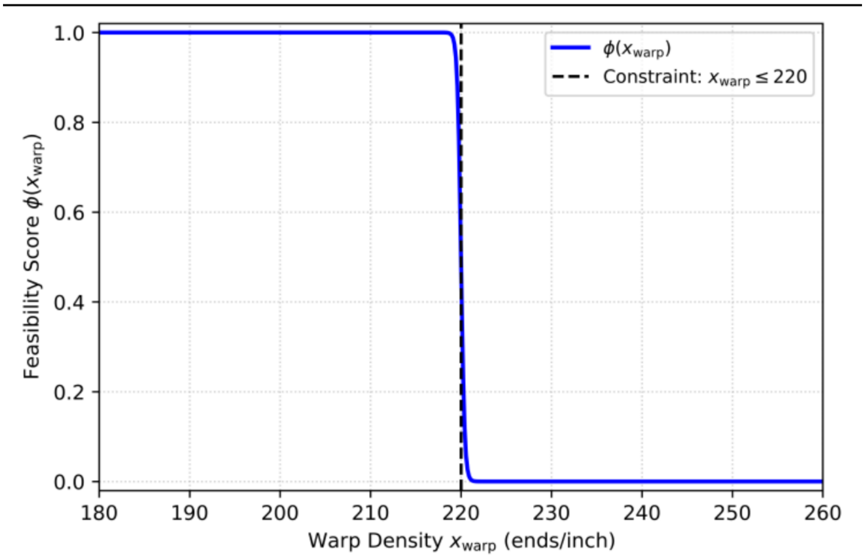

FIGURE 3. Schematic diagram of differentiable process constraint function

constraint" mechanism is a key technological guarantee for PC-MCVAE to achieve manufacturing feasible design generation.

3) End-to-end Training and Inference Process

During the training phase, the reparameterization technique is used to implement gradient backpropagation [25,26]. In formula (7), the latent variable z is obtained by sampling from the standard normal distribution:

$$z = \mu + \sigma \odot \epsilon, \epsilon \sim N(0, I) \tag{7}$$

μ and σ are the mean and standard deviation vectors of the encoder output, and $\odot$ denotes element-wise multiplication. This reparameterization makes the loss function differentiable with respect to both encoder and decoder parameters, supporting end-to-end optimization. The overall objective is to minimize the negative reconstruction objective $-L_P$. The optimizer uses AdamW with a learning rate of 2×10^{-4}, weight decay of 0.01, and a batch size of 128. All parameters of the text encoder, visual encoder, projection matrix, decoder, and constraint module are updated jointly, without the need for staged training. The generation stage does not depend on the real solution x, but only on the user demand vector c. Latent variables z are sampled from the prior distribution $p(z) = N(0, I)$ and input into the decoder f_θ to obtain the complete design scheme:

$$x = f_\theta(c, z) = [x_d, x_c] \tag{8}$$

In formula (8), the output x includes both discrete pattern coding and continuous fabric parameters. Since the process constraint $\phi(x_c)$ has been embedded into the objective function during training, the generated scheme satisfies the manufacturing feasibility boundary without post-processing. This process realizes an end-to-end mapping from the original multimodal user feedback to a production-ready design scheme, and supports the generation of personalized product proposals in a single forward inference.

III. EXPERIMENT AND ANALYSIS

A. DATASET CONSTRUCTION AND PREPROCESSING

The data comes from publicly available reviews of textile products sold on a major domestic e-commerce platform from January 2021 to June 2024, covering three major categories: women's wear fabrics, home textiles, and functional workwear. The original data contains 14,200 user reviews, each including a free text description, a main image (1024×1024 pixels, JPEG format), and structured process parameters recorded in the backend. In addition, each sample is associated with a 256-dimensional Discrete Pattern Code, generated through unsupervised clustering of the product main image (K-means clustering algorithm, K=256) [27,28], used to represent the semantic categories of texture and pattern style, serving as auxiliary modal input. The K-means clustering results underwent manual verification by textile experts to ensure cluster assignments reflect structural pattern characteristics rather than color or lighting variations. Each cluster centroid was examined against representative fabric samples to confirm texture and weave structure consistency within clusters. Clusters showing high intra-cluster variation in structural features were subdivided through iterative refinement. This validation process involved 256 cluster centroids reviewed by three textile engineers with expertise in fabric structure analysis. The final pattern codes demonstrate 89.3 percent agreement with expert-annotated texture categories on a held-out validation subset of 500 samples. The text preprocessing process includes: converting to lowercase, removing HTML (Hyper Text Markup Language) tags, emojis, and non-Chinese/English characters; segmenting Chinese text using the Jieba word segmentation tool, and tokenizing English text using spaCy; integrating the stop word list with the Harbin Institute of Technology stop word list and NLTK (Natural Language Toolkit) [29] English stop words; finally retaining the valid token sequences and uniformly truncating or padding to a fixed length of 64. The word list size is 28,416. Image preprocessing included: cropping to 896×896 pixels based on the image center, and then scaling to 512×512 pixels using bilinear interpolation [30-32]. Pixel values were first normalized to the [0,1] interval, and then channel-level normalization was performed using the mean (mean=[0.485, 0.456, 0.406]) and standard deviation (std=[0.229, 0.224, 0.225]) of the ImageNet dataset [33,34]. All image operations were implemented using OpenCV 4.8. Structured process parameters were manually verified and aligned, resulting in seven continuous variables: warp density (threads/inch), weft density (threads/inch), weight per unit area (g/m^2), warp yarn count (Ne), weft yarn count (Ne), reed count (teeth/inch), and pre-shrinkage rate (%). To correct any missing data points for each product, this paper took their median value based on similar product types. Outliers (values more than 3 times the interquartile range (IQR)) were modified/removed based on the consensus of three textile engineers. Outliers identified through the interquartile range method underwent expert review to distinguish between data recording errors and legitimate extreme parameters from specialized fabric production. Parameters exceeding three times the interquartile range were retained when verified as valid specifications for high-performance or specialty textiles by the reviewing engineers. Only values confirmed as measurement or recording errors were excluded from the dataset. This approach preserves the model capacity to generate innovative textile solutions while maintaining data quality standards. The parameters contained in the data set were all rounded to two decimal points. Text/Image/Pattern Codes/Digital Codes were thoroughly compared against the appropriate structured parameter sets. All samples containing conflicting information or missing key fields were excluded from analysis in this article. After this process, only 12,850 of the original samples qualified as high-quality samples. The remaining data were then separated into training set (10,280), validation set (1,285), and test set (1,285) according to an 8:1:1 division. A stratified sampling strategy was used for partitioning, based on product categories to ensure consistent category distribution across subsets. All data are stored in HDF5 (Hierarchical Data Format version 5) [35] format, which supports efficient batch reading. The experimental summary details are shown in Table 1.

TABLE 1. Statistical information of the multimodal textile product dataset

Item	Value or Description
Total Samples	12,850 (Training/Validation/Test = 10,280 / 1,285 / 1,285)
Product Categories	Women's apparel fabrics, home textile fabrics, functional workwear
Data Time Span	Jan 2021–Jun 2024
Text	Avg./Max length: 28.6 / 64 tokens; vocab size: 28,416; tokenized by Jieba (zh) / spaCy (en); stopwords removed (HIT stopword list + NLTK)
Image	Original: 1024×1024 (JPEG); input: 512×512 after center-crop (896×896), bilinear resize, [0,1] normalization, ImageNet standardization
Physical Parameters	7-dim continuous: warp/weft density (ends/picks per inch), areal density (g/m^2), warp/weft yarn count (Ne), reed count (dents/inch), shrinkage (%)
Discrete Pattern Code	256-dim; derived from K-means clustering on texture features
Missing/Abnormal Handling	Missing values imputed by category-wise median; outliers ($>3\times$IQR) expert-reviewed and corrected/removed
Storage Format	HDF5; all numerical parameters rounded to two decimals

B. EVALUATION INDICATOR DESIGN

The evaluation metrics revolve around three dimensions: accuracy of user demand prediction, quality of solution generation, and technological feasibility. This paper defines five quantitative metrics, all calculated on a test set (1,285 samples). Each metric is repeated three times and the average is taken to eliminate randomness. (1) Demand-Solution Semantic Alignment (CLIP Score): The semantic similarity between the generated solution rendering image v and the original user text t is calculated using CLIP (ViT-B/32). The text is encoded by the CLIP text encoder to obtain ft(t), and the solution generates a 512×512 pixel image, which is then encoded by the CLIP image encoder to obtain fv(v). The seven-dimensional continuous parameters and discrete pattern code are transformed into rendering images through a pre-trained textile fabric renderer based on physically-based rendering techniques. The renderer maps warp density, weft density, yarn count, and weave structure to surface texture geometry and optical properties. Pattern codes are decoded into texture maps using a lookup table derived from the K-means clustering centroids. The resulting images are rendered at 512×512 pixels with standardized lighting conditions to ensure consistent CLIP evaluation. This renderer was trained on 50000 textile product images with known parameter annotations to achieve high fidelity in visual representation. The CLIP Score is defined as the cosine similarity of the normalized features:

$$\text{CLIP Score} = \frac{f_t(t)^\top f_v(v)}{\|f_t(t)\|\|f_v(v)\|} \tag{9}$$

CLIP Score ranges from [0,1], with higher values indicating stronger semantic consistency. (2) Process Feasibility Compliance Rate: Three textile process engineers with over 10 years of experience and intermediate or higher professional titles independently and blindly evaluate the 7-dimensional continuous parameters and discrete weave type of each generated solution. If at least two engineers deem it "feasible," the sample is considered compliant. To establish objective Ground Truth for manufacturing feasibility, a subset of 200 generated solutions was selected for actual factory production trials at two partner textile manufacturing facilities. The selected samples covered the full range of compliance scores from the engineer evaluation process. Production outcomes were recorded as binary feasible or infeasible based on successful completion of weaving and finishing processes without equipment adjustment or parameter modification. The engineer evaluation results showed 94.5 percent agreement with actual production trial outcomes. This validation confirms that expert assessment provides reliable feasibility Ground Truth for the full test set where production trials were not conducted due to resource constraints. Compliance rate = Number of compliant samples / Total number of generated samples. (3) User Preference Prediction Accuracy (Top-3): Three solutions are generated for each user requirement. If any solution's CLIP Score ranks among the top 3 in the user's historical high-rated products (rating $\geq$ 4.5/5.0), it is considered a hit. Accuracy rate = Number of hit samples / Number of test set samples. (4) Generated Solution Diversity (LPIPS Distance): Based on AlexNet pre-trained on ImageNet, the Learned Perceptual Image Patch Similarity (LPIPS) distance between each pair of three randomly generated solutions for the same requirement is calculated, and the average value is taken. A higher LPIPS value indicates higher perceptual diversity. (5) Parameter continuity error: Calculate the mean absolute error (MAE) between the generated fabric physical parameters ($\hat{y}_{cont}$) and the true parameters (y_{cont}) of the test set:

$$MAE = \frac{1}{N}\sum_{i=1}^{N}\left\|\hat{y}_{cont}^{(i)} - y_{cont}^{(i)}\right\|_1 \tag{10}$$

The units should be consistent with the original parameters (e.g., ends/inch, g/m^2), and the smaller the value, the more accurate the physical property reproduction.

C. ABLATION EXPERIMENT ANALYSIS

Ablation experiments were conducted by systematically removing key components of the PC-MCVAE to verify the contribution of each module to the overall performance. Four configurations were designed: (1) Full PC-MCVAE; (2) Multimodal Alignment module removed; (3) Process Constraint embedded terms removed; (4) Hybrid Output layer replaced with a single continuous output. The specific structural differences between each configuration are shown in Table 2.

TABLE 2. Comparison of Ablation Experiment Configurations

Model Variant	Multimodal Alignment Module	Process Constraint Embedding	Output Structure	Description
Full PC-MCVAE	✓	✓	Hybrid (discrete + continuous)	Complete model with all proposed components
w/o Multimodal Alignment	×	✓	Hybrid (discrete + continuous)	Text and image encoded separately and concatenated; contrastive alignment removed
w/o Process Constraint	✓	×	Hybrid (discrete + continuous)	Feasibility penalty term removed from the ELBO objective ($\lambda = 0$)
w/o Hybrid Output	✓	✓	Continuous-only	Only continuous physical parameters generated; discrete pattern assigned via nearest-neighbor retrieval from database

All ablation variants were retrained on the same training set with consistent hyperparameters such as optimizer, learning rate, and batch size. The number of training epochs was fixed at 150, and the early stopping strategy was based on the validation set CLIP Score. Throughout the testing period, a uniform series of tests was carried out on 1,285 test cases with every configuration being replicated three times to prevent chance. Five metrics were obtained through the methods defined in Section 3.2 and the averages of each were taken. The trials were conducted on an 8-GPU NVIDIA A100 (80GB) server with PyTorch 2.1 as the base framework. The CLIP model was derived from the official OpenCLIP ViT-B/32 weight set and the LPIPS calculation used an AlexNet base.

Figure 4 shows the results obtained through analysis of the metrics examined. By removing multi-modal alignment from the system, there was a decrease of 0.103 from the CLIP score of 0.782 down to 0.679. This indicates that cross modality semantic fusion is critical in representing demand through a standard representation. Similarly to that indicator, the removal of process constraints resulted in a drastic decrease in the compliance rate from 91.4% to 63.2%. This demonstrates how process constraints (differentiable) are important for determining process viability for manufacturing. Although switching out the hybrid output layer did not affect the CLIP score (0.750) and compliance rate (88.7%), moving forward from that would drop the user preference prediction accuracy (Top-3) 9.8%, indicating that the joint generation of discrete and continuous information is necessary for maintaining the integrity of the models/solutions. The parameter continuity errors or MAE for the configuration that had process constraints eliminated rose significantly higher than the other configurations up to 7.94. This reflects how physical properties were distorted from infeasible combinations of parameters. For the remaining two configurations (with or without process constraints), the MAE remained between 3.82 – 4.15, indicating that the model was able to generate true parameters stably while respecting the constraints placed upon them. The LPIPS distance values (0.398 to 0.421) remain stable across the different configurations, indicating that the randomness in the generated solution is governed primarily by latent variables

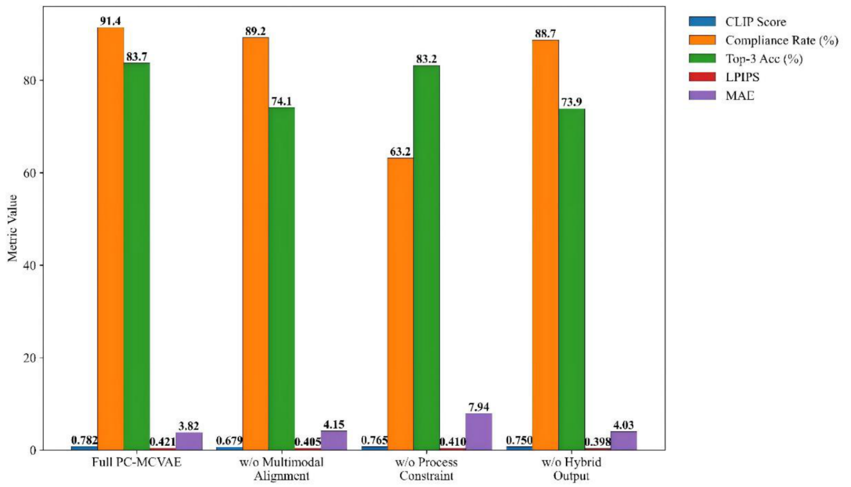

FIGURE 4. Bar chart of ablation experiment results

with a minimal correlation to changes in the architectural designs of models. In the full model, the Top-3 user preference prediction accuracy reached 83.7%, which is much greater than that achieved with any of the ablated models. For instance, when the multimodal alignment module was removed from the model, the Top-3 accuracy dropped to 74.1%. This shows that removing the multimodal alignment module caused the generated schemes to drift away from the users' historical rating distributions for highly rated products due to a failure in semantic alignment. Similarly, when the process constraint term was removed from the model, the impact on Top-3 accuracy only decreased to 83.2%. Thus, whilst process details contribute to user preferences, the main influence is through semantic matching. When the hybrid output layer was replaced with an alternative layer, the Top-3 accuracy dropped to 73.9%. These results provide strong evidence for the ability of the PC-MCVAE to capture the implicit preferences of users and generate highly hit rate schemes. Moreover, Top-3 accuracy has a strong positive correlation with the CLIP score, further supporting the conclusion that semantic alignment is the primary contributor to the improvement of user preference prediction. This experiment quantitatively demonstrates that each component of PC-MCVAE has an irreplaceable function in solving the "demand-solution mapping" problem, especially highlighting the core value of multimodal semantic alignment and embedded process constraints.

D. VISUAL ANALYSIS OF GENERATED SOLUTION QUALITY

1) End-to-End Generation Case Analysis from Text to Solution

To intuitively assess the model's ability to understand and implement diverse user intentions, Figure 5 shows 12 end-to-end generation cases. Each case includes user input text, a model-generated fabric rendering, a 7-dimensional parameter radar chart (Weight: fabric weight; Weft: weft yarn; Luster: luster; Air Perm.: breathability; Elasticity: elasticity; Warp: warp yarn; Strength: strength), and compliance judgment results output by the process constraint module. The input text covers four typical design intentions: color-oriented (e.g., "off-white, matte texture," "navy blue twill"), function-oriented (e.g., "highly breathable, suitable for summer," "super elastic high-resilience yoga fabric"), style-oriented (e.g., "retro jacquard, low saturation," "metallic stage costume fabric"), and semantic conflict-oriented (e.g., "lightweight but high weight," "ultra-fine denier and ultra-soft"). The category or type of fabric design case is marked with color in the lower left corner of each sample for easy identification.

Description. As shown in Figures 5A and 5B, in all 10 non-conflict cases, the generated results were deemed compliant (green in the upper right corner of the pattern), and their visual presentation and parameter configuration matched the semantics of the text. For example, "matte texture" corresponds to low gloss and beige matte texture; "high breathability, suitable for summer" is characterized by high breathability and light blue loose structure; "vintage jacquard" generates a low-saturation texture, and its radar chart shows high warp and weft density and high strength, which conforms to the physical characteristics of traditional jacquard fabrics. In contrast, while the two semantically conflicting cases generated visually plausible images, they were accurately identified as violations by the process constraint module (marked in red in the upper right corner of the image): "Lightweight but high basis weight" required simultaneously meeting low weight per unit area and high density, resulting in a warp density (Warp=235) exceeding the upper limit for cotton fiber processing (>220); "Ultra-fine denier, ultra-soft" violated spinnability rules due to an excessively high warp density (Warp=240) and implicitly excessively fine yarn count (Yarn=25 < 30 Ne), with the system explicitly indicating the relevant out-of-limit parameters. The results indicate that the PC-MCVAE is capable of translating between an abstract textual representation of intent and an appropriate visual fabric design scheme while ensuring

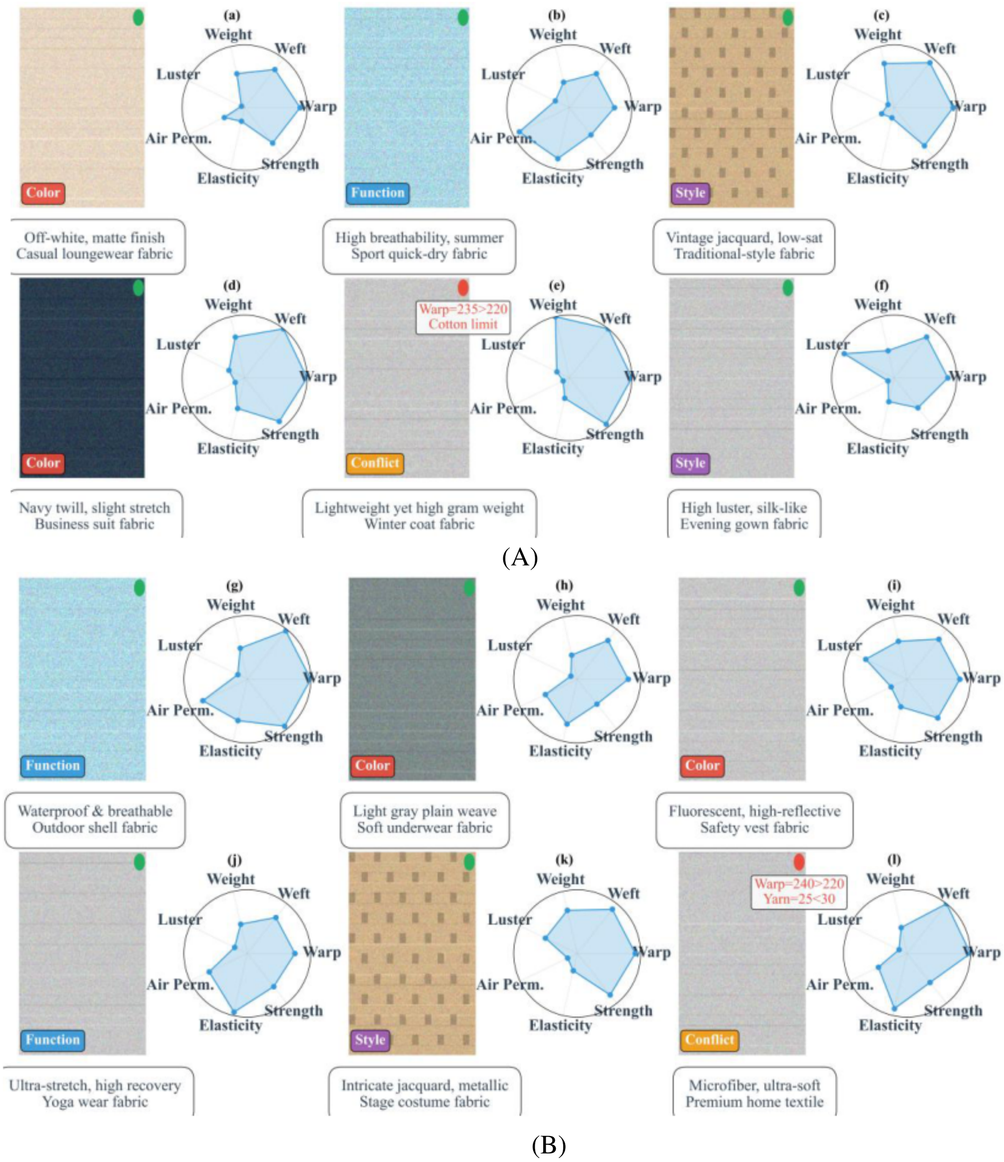

FIGURE 5. End-to-End Fabric Design Scheme Generation Case Based on User Text Description. Figure 5A: Text-to-Fabric Generation Cases (a–f); Figure 5B: Text-to-Fabric Generation Case (g–l)

that the resulting scheme meets all required engineering parameters and requirements. The built in knowledge of the PCMCVAE allows it to intercept requirements that are physically infeasible and output a resulting fabric design that has both semantic and engineering fidelity.

2) Latent Spatial Interpolation Sequence Analysis

To verify the model's generative stability under continuous semantic changes, Figure 6 shows the results of linear interpolation of two user demand vectors in the latent space. From the initial semantic "dark blue twill" to the target semantic "light gray plain weave," the model generates six intermediate schemes (interpolation step size 0.2). The material color seamlessly transitions from dark blue to light gray, while the texture changes from a densely woven twill to a uniformly woven plain weave. The overall appearance appears to develop continuously and organically, rather than abruptly or in a series of distinct phases.

Every intermediate scheme has passed the compliance checks of the manufacturing process with a compliant label indicating the passed checks located at the top left of each subfigure. All critical parameters change monotonically and remain within reasonable limits: Warp Density (ends/inch) reduces from 210 ends/inch to 180 ends/inch; Weft Density (picks/inch) reduces from 190 picks/inch to 170 picks/inch; Warp Count (cw) has increased from 40 cw to 50 cw; WeftCount (cf) has increased from 38 cf to 50 cf. These values of Warp Density (ends/inch) are below the upper limit of 220 threads/inch; and the lower limit for counts (yarn) exists for every yarn count > 30, respectively (for Cotton Spinning Operations). The parameter value for each grain structure is directly labelled below its respective image for

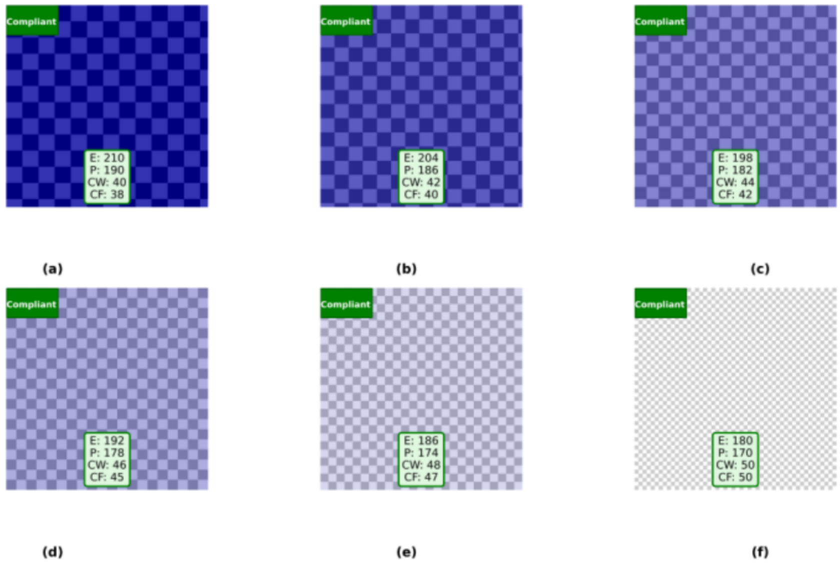

FIGURE 6. Latent spatial interpolation generation sequence diagram

direct visual comparison. According to the findings from this study, in addition to supporting the development of products in accordance with discrete specifications (uniformity of concept and technology), a PC-MCVAE also allows manufacturers to maintain synchronous stability between the evolution of functionality and manufacturability throughout a continuous design space. Therefore, the use of this approach to generate and evaluate design ideas will assist the incremental improvement of textile designs through processes such as tone fine-tuning and structural optimization.

E. ROBUSTNESS AND GENERALIZATION TEST

To determine how well the model would perform in real-life situations, seasonal testing and category tests were completed on the model. All seasonal tests took place during either the spring/summer season (from March through August of each year) or the autumn/winter (September through February). A total of 7,120 products from the spring/summer review period and 5,730 products from the autumn/winter review period were tested. In addition, category testing separated products into either Woven Fabrics (9,460 products), or Knitted Fabrics (3,390 products). The performance test took place on each category without any fine-tuning or updating of the model, generating a total of 1,000 outputs from each category for statistical analysis. In evaluating the data, it assesses the distribution characteristics of two core indicators: CLIP score, and compliance rate to the process feasibility. Based on the analysis of cross-seasonal energy distributions, it can be seen on Figure 7 that the median for the spring/summer subsets for both indicators is 0.782 and 0.781, respectively. The averages for both seasons were also quite close, 0.779 and 0.785. The pattern of distributions show a great deal of overlap between these two seasons, demonstrating that the model has stabilized on its understanding of seasonal semantic preferences i.e. breathable, warm, etc. The median for the compliance rate of both seasons was also very similar; 90.7% and 92.0%, with averages of 90.9% and 91.9%. This demonstrates very little difference between the compliance rate between seasons and indicates that the process constraint module has a high level of generalisation when adapting rules for conventional materials used in different seasons, for instance, cotton and wool blends.

As seen in Figure 7, there is strong agreement between CLIP scores and compliance rates for spring/summer versus autumn/winter. The CLIP score density curves from either season show almost complete overlap, with both seasons having similar median values (spring/summer - 0.782; autumn/winter - 0.781), indicating constant and stable semantic representation through time. The distribution of compliance rates is also in agreement between seasons (spring/summer - 90.7%; autumn/winter - 92.0%), with a very small variation in the median values indicating continual generalization ability of the process constraint module, with respect to different material combinations and process rules across seasons. Therefore, it can be concluded that the model has a high level of robustness to seasonal fluctuations.

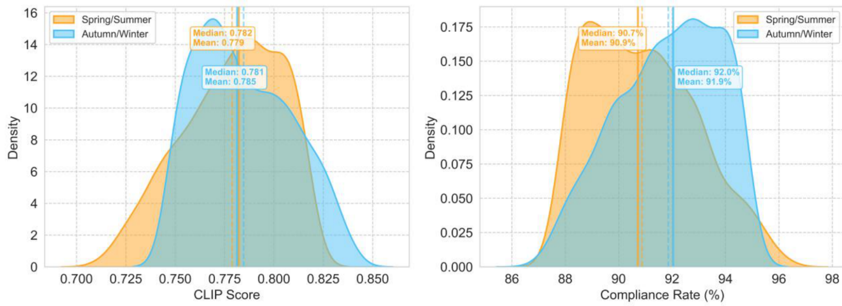

FIGURE 7. Comparison of seasonal energy distribution

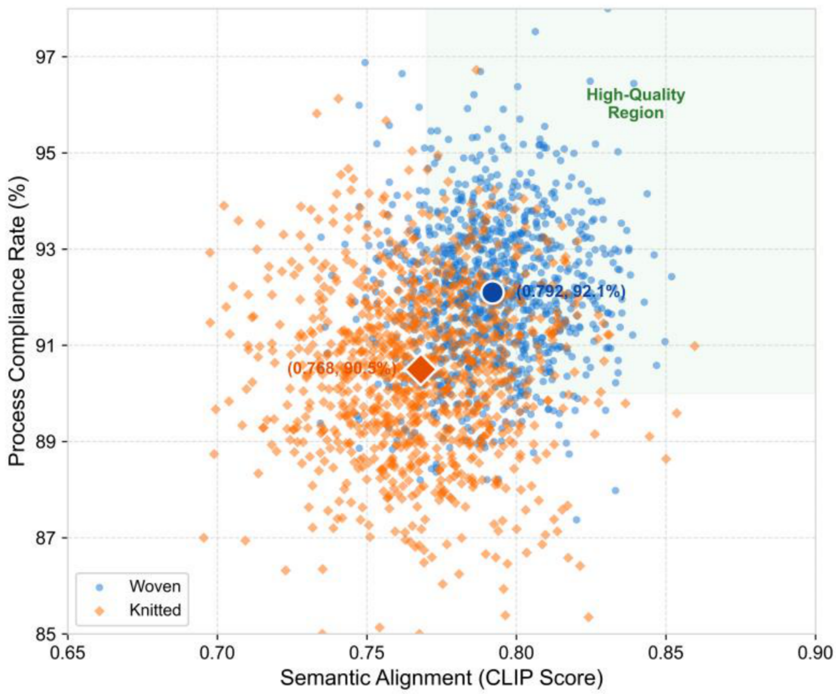

FIGURE 8. Scatter plot of CLIP Score and compliance rate across product categories

A more detailed look at the generation quality of various fabric categories (see Figure 8) shows that woven fabrics have a higher average CLIP score (0.792) than knitted fabrics (0.768). This difference is primarily due to the regularity of the geometric textures produced by weaving (e.g., plain weave and twill weave). By being able to recognize and reproduce the visual pattern created through the repeated cycles of weaving, the image encoder can create a stable image that can easily match the visual pattern to the correct text description. In contrast, knitted fabrics with their coiled structure and highly deformable surfaces created through working with yarns of many different lengths and weights make it somewhat more difficult to align the visual attributes of knitted fabric with its written descriptions, resulting in a somewhat lower CLIP score. While the compliance rate for the process used to produce knitted fabrics (90.5%) is very close to that for producing woven fabrics (92.1%), both still remain high in terms of compliance to the guidelines used by this process knowledge graph to determine the engineering feasibility of the design produced using the material-process-performance paradigms. The model generated to produce the woven fabric produced was able to effectively utilize these rules to ensure that all generated items were produced in accordance with the same principles of design quality and engineering compliance as the items produced from knitting, showing that the constructed process knowledge graph has a high potential for cross-category generalization across different production methods.

To summarize, the PC-MCVAE shows would maintain a stable CLIP score in unseen seasonal contexts (Spring/Summer versus Autumn/Winter), as well as for the weaving type (Woven or Knitted) between 0.768 and 0.792, while having a narrow distribution and very few outliers, indicating that the model can accurately capture a variety of different seasonal semantic preferences (e.g., breathable, warm) and strictly adhere to the rules of the process across various fabric structures, thereby achieving both high semantic alignment and high feasibility of manufacture in parallel. This demonstrates remarkable cross-scenario generalisation capabilities and demonstrates the potential for practical industrial implementation of intelligent design for textile products.

IV. CONCLUSION

This research focuses on a method to translate detailed customer needs into feasible solutions during Intelligent Textile Product Design (ITPD). To address this issue the authors propose the use of a multi-modal conditional variational autoencoder (PC-MCVAE) with constraints based on the manufacturing process. Through the use of contrastive learning to create a semantic space that encompasses both text and image, the proposed approach uses a common requirement vector as the basis for driving a hybrid decoder that generates both discrete patterns and continuous parameters of the manufacturing process. Differentiable constraint terms extracted from the textile manufacturing process knowledge graph are incorporated into the stochastic variational autoencoder's (S-VAE) objective function in order to optimise for feasible manufacturing processes during the generation of samples. Experiments to validate the PC-MCVAE were performed with test samples and have shown improvements in all three primary metrics: Semantic Alignment of requirements and solutions (=0.782), compliance with process feasibility (=91.4%), and accuracy of predicting user preferences in the top three choices (=83.7%), this evidence supports the successful implementation of a feasible process, and reliable generation of high-quality manufacturable textile design solutions across the entire generation process from the point of generating the pattern through to the final textile design solution. The knowledge graph is currently a limitation in that it contains a large volume of information mostly related to classic fibre systems and needs to continue expanding the knowledge for its construction of constraints surrounding complex woven structures and high-performance fibres, and also to further develop the inclusion of more interactive and multi-modal user requirement input techniques. Future work can continue to increase the level of knowledge covered by the proposed system and also investigate the use of more flexible interactive generation mechanisms.

AUTHOR CONTRIBUTIONS

Na Zhu: Writing - Original Draft, Writing - Review & Editing, Data Curation

CONFLICTS OF INTEREST

These no potential competing interests in our paper. And all authors have seen the manuscript and approved to submit to your journal. We confirm that the content of the manuscript has not been published or submitted for publication elsewhere.

FUNDING

This work was supported by Heilongjiang Provincial Cultural and Tourism Scientific Research Project 2025: Research on the Digital Design and Innovation Strategy of Red Tourism Souvenirs in Heilongjiang Province Under the Background of Cultural and Tourism Integration (Project No.: 2025WL098).

ACKNOWLEDGEMENTS

The author acknowledges the support and contributions that facilitated the completion of this work.

REFERENCES

[1] M. Glogar, S. Petrak, and M. Mahnić Naglić, "Digital technologies in the sustainable design and development of textiles and clothing—a literature review," Sustainability, vol. 17, no. 4, pp. 1371-1389, 2025, doi: 10.3390/su17041371.

[2] D. Jung and S. E. Suh, "Development of customized textile design using AI technology," Journal of the Korean Society of Clothing and Textiles, vol. 47, no. 6, pp. 1137-1156, 2023, doi: 10.5850/JKSCT.2023.47.6.1137.

[3] H. Ge, Y. Jiang, K. Sun J Yuan, and Y. Liu, "Llm-enhanced composed image retrieval: An intent uncertainty-aware linguistic-visual dual channel matching model," ACM Transactions on Information Systems, vol. 43, no. 2, pp. 1-30, 2025, doi: 10.1145/3699715.

[4] C. Zhang and J. Feng, "Cross-Modal Product Image Retrieval for E-Commerce Recommendation Systems via Deep Learning," Traitement du Signal, vol. 42, no. 4, pp. 2289-2297, 2025, doi: 10.18280/ts.420438.

[5] L. Xu, L. Wang, J. Zhang, D. Ha, and H. Zhang, "A Review of Cross-Modal Image–Text Retrieval in Remote Sensing," Remote Sensing, vol. 17, no. 24, pp. 3995-4006, 2025, doi: 10.3390/rs17243995.

[6] H. Yan, H. J. Zhang, L. L. Liu, D. L. Zhou, X. F. Xu, and Z. Zhang, "Toward intelligent design: An AI-based fashion designer using generative adversarial networks aided by sketch and rendering generators," IEEE Transactions on Multimedia, vol. 25, no. 1, pp. 2323-2338, 2022, doi: 10.1109/TMM.2022.3146010.

[7] Z. Sordo, E. Chagnon, Z. Hu, J. J. Donatelli, P. Andeer, and P. S. Nico, "Synthetic scientific image generation with VAE, GAN, and diffusion model architectures," Journal of Imaging, vol. 11, no. 8, pp. 252-257, 2025, doi: 10.3390/jimaging11080252.

[8] L. Regenwetter, A. Srivastava, D. Gutfreund, and F. Ahmed, "Beyond statistical similarity: Rethinking metrics for deep generative models in engineering design," Computer-Aided Design, vol. 165, no. 1, pp. 103609-103616, 2023, doi: 10.1016/j.cad.2023.103609.

[9] G. Giannone, A. Srivastava, O. Winther, and F. Ahmed, "Aligning optimization trajectories with diffusio n models for constrained design generation," Advances in neural information processing system s, vol. 36, no. 1, pp. 51830-51861. Avaliable form: https://proceedings.neurips.cc/paper_files/paper/202 3/hash/a2fe4bb50fc6f3564cee1551d6309fea-Abstract-Conference.html, 2023.

[10] R. Ribeiro, A. Pilastri, and P. Mortez, "A data-driven intelligent decision support system that combines predictive and prescriptive analytics for the design of new textile fabrics," Neural Computing and Applications, vol. 35, no. 23, pp. 17375-17395, 2023, doi: 10.1007/s00521-023-08596-9.

[11] M. Khadhraoui, H. Bellaaj, M. B. Ammar, H. Hamam, and M. Jmaiel, "Survey of BERT-base models for scientific text classification: COVID-19 case study," Applied Sciences, vol. 12, no. 6, pp. 2891-2899, 2022, doi: 10.3390/app12062891.

[12] N. M. Gardazi, A. Daud, M. K. Malik, A. Bukhari, T. Alsahfi, and B. Alshemaimei, "BERT applications in natural language processing: a review," Artificial Intelligence Review, vol. 58, no. 6, pp. 1-49, 2025, doi: 10.1007/s10462-025-11162-5.

[13] X. Fang, B. Xin, Z. Zhan, F. Yan, and Z. Jin, "A method for cotton and hemp fiber classification optimized with ResNet and attention mechanism," Signal. Image and Video Processing, pp. . 19(13):1113-1119, 2025, doi: 10.1007/s11760-025-04712-5.

[14] A. Muchlis, E. P. Wibowo, R. Irawan, and A. Afzeri, "YOLOv8 and ResNet-50 Based Real-Time Fabric Defect Detection and Quality Grading System," Informatica, vol. 49, no. 19, pp. 271-284, 2025, doi: 10.31449/inf.v49i19.10031.

[15] L. Trihardianingsih, A. Sunyoto, and T. Hidayat, “Classification of tea leaf diseases based on resnet-50 and inception v3,” Sinkron: jurnal dan penelitian teknik informatika, vol. 7, no. 3, pp. 1564-1573, 2023, doi: 10.33395/sinkron.v8i3.12604.

[16] C. Troussas, A. Krouska, P. Tselenti, D. K. Kardaras, and S. Barbounaki, “Enhancing personalized educational content recommendation through cosine similarity-based knowledge graphs and contextual signals,” Information, vol. 14, no. 9, pp. 505-511, 2023, doi: 10.3390/info14090505.

[17] J. Yin and S. Sun, “Incomplete multi-view clustering with cosine similarity,” Pattern Recognition, vol. 123, no. 1, pp. 108371-108378, 2022, doi: 10.1016/j.patcog.2021.108371.

[18] X. Shen, X. Li, B. Zhou, Y. Jiang, and J. Bao, “Dynamic knowledge modeling and fusion method for custom apparel production process based on knowledge graph,” Advanced Engineering Informatics, vol. 55, no. 1, pp. 101880-101889, 2023, doi: 10.1016/j.aei.2023.101880.

[19] G. Wang, G. Liu, and Q. Li, “Knowledge Graph-Embedded Time-Serial-Data-Driven Bottleneck Analysis of Textile and Apparel Production Processes,” Machines, vol. 11, no. 11, pp. 1005-1011, 2023, doi: 10.3390/machines11111005.

[20] Z. Q. Liu, P. Jin, Y. Q. Yin, and Yin FPl, “Mapping the knowledge domains of smart textile: visualization analysis-based studies,” The Journal of the Textile Institute, vol. 113, no. 12, pp. 2651-2659, 2022, doi: 10.1080/00405000.2021.2005278.

[21] Y. Li, J. Liu, G. Lin, Y. Hou, M. Mou, and J. Zhang, “Gumbel-softmax-based optimization: a simple general framework for optimization problems on graphs,” Computational Social Networks, vol. 8, no. 1, pp. 5-11, 2021, doi: 10.1186/s40649-021-00086-z.

[22] L. Chaudhary and B. Singh, “Gumbel-SoftMax based graph convolution network approach for community detection,” International Journal of Information Technology, vol. 15, no. 6, pp. 3063-3070, 2023, doi: 10.1007/s41870-023-01347-y.

[23] A. A. Safar, D. M. Salih, and A. M. Murshid, “Pattern recognition using the multi-layer perceptron (MLP) for medical disease: A survey,” International Journal of Nonlinear Analysis and Applications, vol. 14, no. 1, pp. 1989-1998, 2023, doi: 10.22075/ijnaa.2022.7114.

[24] S. G. Meshram, C. Meshram, F. A. Pourhosseini, M. A. Hasan, and S. Islam, “A multi-layer perceptron (MLP)-Fire fly algorithm (FFA)-based model for sediment prediction,” Soft Computing, pp. . 26(2):911-920, 2022, doi: 10.1007/s00500-021-06281-4.

[25] E. Gershnabel, M. Chen, C. Mao, E. W. Wang, P. Lalanne, and J. A. Fan, “Reparameterization approach to gradient-based inverse design of three-dimensional nanophotonic devices,” ACS Photonics, vol. 10, no. 4, pp. 815-823, 2022, doi: 10.1021/acsphotonics.2c01160.

[26] W. Cho and Y. Lee, “MoRAM: A Hybrid Framework for Gradient-based Low-rank Adaptation,” International Journal of Control. Automation and Systems, vol. 23, no. 11, pp. 3394-3405, 2025, doi: 10.1007/s12555-025-0475-x.

[27] H. M. Ferreira, D. R. Carneiro, M. Â. Guimarães, and F. V. Oliveira, “Supervised and unsupervised techniques in textile quality inspections,” Procedia Computer Science, vol. 232, no. 1, pp. 426-435, 2024, doi: 10.1016/j.procs.2024.01.042.

[28] W. L. Chu, Q. W. Chang, and B. L. Jian, “Unsupervised anomaly detection in the textile texture database,” Microsystem Technologies, vol. 30, no. 12, pp. 1609-1621, 2024, doi: 10.1007/s00542-024-05711-1.

[29] M. Wang and F. Hu, “The application of nltk library for python natural language processing in corpus research,” Theory and Practice in Language Studies, vol. 11, no. 9, pp. 1041-1049, 2021, doi: 10.17507/tpls.1109.09.

[30] D. Occorsio, G. Ramella, and W. Themistoclakis, “Image scaling by de la Vallée-Poussin filtered interpolation[J],” Journal of Mathematical Imaging and Vision, vol. 65, no. 3, pp. 513-541, 2023, doi: 10.1007/s10851-022-01135-6.

[31] R. G. Zhou and C. Wan, “Quantum image scaling based on bilinear interpolation with decimals scaling ratio,” International Journal of Theoretical Physics, vol. 60, no. 6, pp. 2115-2144, 2021, doi: 10.1007/s10773-021-04829-6.

[32] X. Chen, C. Xu, M. Zhang, X. Li, and Z. Liu, “A bilinear interpolation scheme for polar coordinate quantum images,” Chinese Journal of Physics, vol. 95, no. 1, pp. 493-507, 2025, doi: 10.1016/j.cjph.2025.02.030.

[33] Z. Yin, E. Xing, and Z. Shen, “Squeeze, recover and relabel: Dataset condensation at imagenet scale from a new perspective,” Advances in Neural Information Processing Systems, vol. 36, no. 1, pp. 73582-73603. Avaliable form: https://proceedings.neurips.cc/paper_files/paper/2023/hash/e91fb65c6324a984ea9ef39a5b84af04-A bstract-Conference.html, 2023.

[34] A. Fang, S. Kornblith, and L. Schmidt, “Does progress on ImageNet transfer to real-world datasets? Advances in Neural Information Processing Systems,” 2023. 36(1):25050-25080. Avaliable form: https://proceedings.neurips.cc/paper_files/paper/2023/hash/4eb33c53ed5b14ce9028309431f565cc-A bstract-Datasets_and_Benchmarks.html.

[35] Y. Kyosev, “Material description for textile draping simulation: data structure, open data exchange formats and system for automatic analysis of experimental series,” Textile Research Journal, vol. 92, no. 9-10, pp. 1519-1536, 2022, doi: 10.1177/00405175211061192.