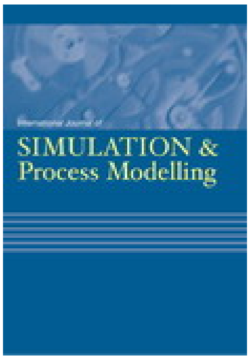
Inderscience Publishers
Linking academia, business and industry through research



International Journal of Simulation and Process Modelling

ISSN online: 1740-2131 - ISSN print: 1740-2123
https://www.inderscience.com/ijspm

Simulation enhanced early warning and intervention model for customer success in startups using value alignment

Hong Chen, Zuwei Yu, Shangui Hu

DOI: 10.1504/IJSPM.2026.10079834

Article History:

Received:	04 January 2026
Last revised:	06 May 2026
Accepted:	26 May 2026
Published online:	14 August 2026

Simulation enhanced early warning and intervention model for customer success in startups using value alignment

Hong Chen

College of Business Administration,
Ningbo University of Finance and Economics,
Ningbo 315175, Zhejiang, China
and
Research Center for the Regional Comprehensive Economic Partnership (RCEP) Member States,
Ningbo 315175, Zhejiang, China
Email: maomao821215@126.com

Zuwei Yu*

Department of Economics and Trade,
Guangdong Eco-Engineering Polytechnic,
Guangzhou 510520, Guangdong, China
Email: yuzuwei@hotmail.com
*Corresponding author

Shangui Hu

College of Adult and Continuing Education,
Ningbo University of Finance and Economics,
Ningbo 315175, Zhejiang, China
Email: stanleyhu@nbufe.edu.cn

Abstract: Subscription startups face high churn and resource limits, demanding efficient customer success solutions. We propose a simulation-driven, intelligent framework that models the customer value alignment process. This framework converts business goals into value intent vectors and leverages a process model to simulate customer behaviour and intervention outcomes. Using graph neural networks, we measure value alignment between behaviour and goals, integrating a counterfactual intervention generator with combinatorial optimisation for dynamic action planning. Evaluated on 12,850 real customers, our model achieves 0.904 AUC and 78.4% 30-day retention, outperforming gradient boosting and LSTM baselines. Simulation results also demonstrate a reduction in calibration error and a cut in customer success manager intervention rates by 18%. This framework enhances early-warning timeliness and intervention efficacy, enabling a rapid shift from behavioural alerts to preemptive actions. Ultimately, it delivers a high-ROI, intelligent solution tailored for resource-constrained startups, balancing predictive accuracy with actionable effectiveness to mitigate churn effectively.

Keywords: customer success; value alignment; graph neural network; counterfactual intervention; early warning system.

Reference to this paper should be made as follows: Chen, H., Yu, Z. and Hu, S. (2026) ‘Simulation enhanced early warning and intervention model for customer success in startups using value alignment’, *Int. J. Simulation and Process Modelling*, Vol. 23, No. 7, pp.1–16.

Biographical notes: Hong Chen received her Master’s degree from Southeast University, P.R. China. Currently, she is currently an Associate Professor in the School of Management, Ningbo University of Finance and Economics. Her research interests include artificial intelligence, emerging industries, and innovation and entrepreneurship.

Zuwei Yu received his MS in Economics in 2006 and PhD in Western Economics in 2010 from Wuhan University, Wuhan, P.R. China. He is currently an Associate Professor with the School of Economics and Trade, Guangdong Eco-Engineering Polytechnic, Guangzhou, P.R. China. His research interests include economic growth, human capital accumulation and technological innovation.

Shangui Hu is a Professor at the College of International Economics and Trade, Ningbo University of Finance and Economics. He received his Doctoral in Management from University of Science and Technology of China. His research interests include cross-cultural studies, information management, e-commerce and human resources development. He has published dozens of papers at top journals including information technology and people, electronic commerce research and applications, international journal of intercultural relations, industrial management and data systems and other international conferences.

1 Introduction

With the rapid development of society, the business environment is highly competitive and iterates rapidly. At present, startups are emerging rapidly as the core driving force for the business environment, technological innovation and economic development. However, a large number of statistical data show that more than 70% of startups face operational challenges after their establishment. Among them, customer churn, insufficient product-market fit and customer management are key factors leading to the failure of startups, especially on the basis of the business model of customised technology services (Zhu et al., 2022). Customer retention rate and customer lifetime value have become core indicators for measuring business operations (Li, 2021). Therefore, in this context, customer success is not only an extension of customer service, but also a strategic function that runs through the product life system.

Customer success management relies heavily on human judgement, as well as regular customer follow-ups or satisfaction surveys. Data is used to capture subtle changes in customer behaviour and potential intervention signals. In this context, when customer problems have accumulated to a stage of significant change, such as complaints, downgrades, or cancellations (Wang and Wu, 2025), the best intervention window has been missed, and the cost of recovery is high and the success rate is low. Therefore, how to identify risks in advance through data-driven methods and proactively intervene to provide personalised support to users before customers have clearly expressed their dissatisfaction and have not yet realised their own usage obstacles has become a core means for startups to improve customer retention, optimise resource allocation, and enhance market competitiveness (Cao et al., 2022).

In recent years, with the rapid development of artificial intelligence technology, machine learning, natural language processing, predictive analysis and other technologies have been widely applied in many fields (Triando et al., 2024). Some studies have applied artificial intelligence technology to the automatic extraction of massive customer data. Identify abnormal customer data and make early interventions. Build a customer health score by analysing multi-dimensional real-time data such as product usage logs, customer work orders, billing behaviour, interaction frequency, and social media interaction (Staub et al., 2023). At the beginning of the emergence of customer risk, an automated or semi-automated intervention strategy is implemented. This model has fundamentally changed the interaction paradigm between customers and enterprises, from the passive response of enterprises to the active empowerment and intervention of enterprises.

Therefore, for startups with limited human and customer resources, creating a customer success early warning system that can perform dynamic analysis of return on investment is of great significance. Startups lack the funds and manpower to build the dedicated data analysis team and CRM system required by the enterprise. A lightweight AI-driven customer success early warning system can provide a low-threshold, integrated system to support the development of startups (Hwang et al., 2025). On the other hand, for startups, the customer base is small, but the value density is high. Therefore, customer churn can easily cause changes in the cash flow and reputation of startups (Paeplow et al., 2025). Therefore, for early intervention and early warning systems for startups, not only should they have strong robustness in terms of early warning capabilities, but they should also consider implementation costs, flexibility and multiple deployment factors.

Although there have been many studies on customer churn prediction, most of these studies focus on finance, telecommunications and other fields, and often have strict model assumptions and feature engineering processes. Moreover, the intervention logic of these models cannot be transferred to general industries for implementation (Gu, 2022). On the other hand, the current early warning system and intervention system are often separate and cannot achieve closed-loop verification. In startups, product iterations are rich, customer journeys are unclear and value migration paths are complex, so this model is required to have stronger adaptability and stronger context extraction capabilities (Shen et al., 2025). For example, if a customer's login frequency drops sharply in a short period of time, it may be because of product problems that lead to poor user experience, or it may be because the customer has completed a core task and entered a stable usage period (Cimino et al., 2025). In this context, the model needs to dynamically combine product change functions and user's historical behaviour to make a comprehensive judgement, rather than simply using dynamic threshold rules, etc.

Therefore, this paper aims to construct a customer success rate early warning and intervention model that combines artificial intelligence analysis. This model employs a lightweight machine learning approach and forms a closed-loop intelligent optimisation system through a configurable intervention engine. We focus not only on the accuracy of predictions but also on the model's operability, interpretability, robustness, and continuous learning capabilities in real-world tasks.

2 Literature review

2.1 Early warning mechanism for customer success

With the rapid rise of the subscription economy and the development of SaaS, customer success has become a core indicator to ensure that customers demonstrate business value during product use or service. Chattopadhyay et al. (2025) explained in their research that the key to customer success and traditional customer service is that customer success has a certain degree of foresight and value orientation. That is to say, enterprises need to actively guide customers to achieve certain goals, rather than passively responding to customer requests. Therefore, on this basis, Di Giannantonio et al. (2022) and Ding (2021) proposed the indicator of customer health in their research. This indicator comprehensively evaluates the customer's status through multiple parameters such as the frequency of software or service use, the adoption rate of functions, and the number of support requests, thereby providing a basis for the next step of intervention. However, these early models all rely on static rules or manual threshold judgements, and cannot perform real-time analysis and comprehensive capture of dynamic behaviours.

On the other hand, with the rapid development of data-driven deep learning methods, customer success management has become an important part of predictive analytics. In the study, a framework for analysing customer churn using data mining was proposed, which was mainly implemented in telecommunications web pages (Zhang, 2024). However, the core concept of the method has provided some information foundation for subsequent theories, and the intention is to systematically apply logistic regression neural networks and decision tree-related models in the study to the sub-task of churn prediction, and the experimental results show that different models have large differences in interpretability and prediction accuracy. Takas et al. (2024) constructed a customer success lifecycle model in the study, which divides the customer's product use process into multiple different stages such as activation, growth, and maturity, and constructs differentiated early warning mechanisms and intervention strategies for different stages. However, these studies still face certain difficulties in the face of startups with imperfect organisational structures and rapid product iteration. Therefore, how to construct an early warning and intervention mechanism in such an environment has become a new research direction.

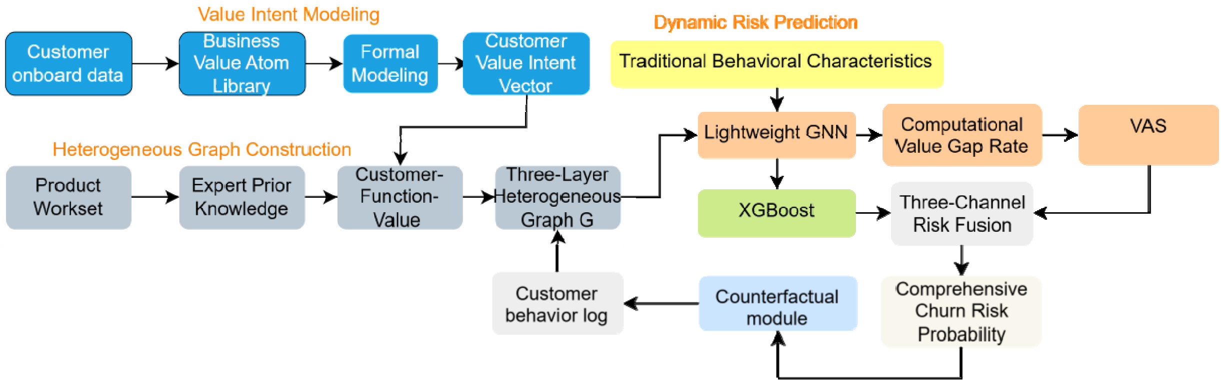
2.2 Artificial intelligence approaches for customer behaviour prediction

Deep learning: there is a lot of research on customer behaviour prediction. Li and Qin (2025) first used ensemble learning, including gradient boosting trees and random forests, in the research. Empirical studies were conducted on multiple industry datasets, and the experimental results showed that feature engineering plays a key role in the effectiveness of different models. The features extracted in the study include multiple indicators such as session duration, function usage diversity, error log frequency, and behavioural features. On the other hand, Algawani and Alsabry (2025) constructed a framework model combining graph neural networks and long short-term neural networks in the research to capture the complex dependencies of customers in the product usage path. This model significantly improves the ability to identify and warn of the churn of silent customers. Giuggioli et al. (2025) further combined natural language processing technology in the research, conducted sentiment analysis and intent recognition framework on the text in the customer usage data, and used the recognition results as the input variables of the warning model, effectively supplementing the information missing in the traditional method. Bessen et al. (2022) combined online learning in the research to enable the model to dynamically update the model as customer behaviour data continues to flow in. This approach can address the training data drift problem caused by rapid product iteration in some startups. These studies all indicate that most current work focuses on the prediction itself, without considering how to translate the prediction results into actionable interventions, especially in the context of startups. How can we build a lightweight solution that integrates prediction and intervention?

2.3 Exploration of intervention strategies

If only an early warning mechanism is implemented, customer retention cannot be guaranteed. Therefore, we must form a closed loop of intervention and early warning. Wang and Guan (2025) constructed a pyramid model for customer retention in their research. This model divides intervention policies into several levels, including automated outreach, personalised communication, and high-level intervention. The three-level model will automatically match different intervention intensities according to the risk level. Based on this research, Raneri et al. (2023) further constructed a framework for optimising intervention timing in their research. Based on the previous research, they considered the optimisation of intervention timing. They combined reinforcement learning to dynamically select the best intervention starting point to avoid over-intervention or intervention lag. Similarly, Li et al. (2025) also constructed an intervention mechanism that combines causal judgement in their research. This mechanism uses counterfactual analysis to analyse whether customer retention is the result of effective intervention or natural retention, thereby dynamically optimising the intervention policy library.

Figure 1 Model framework (see online version for colours)



3 Customer success intelligent early warning and intervention model

3.1 Frame structure

As shown in Figure 1, in the intelligent business system of SaaS, whether customers can use the product further depends on whether the product can bring customers the business value they expect, which we define as EBV. However, in the traditional model, the key indicator is behavioural activity. For example, the decrease in login frequency, the reduction in function calls, etc. (Aziz et al., 2025). However, such practices are often based on specific assumptions, such as that users will definitely get value by using the product. However, in actual use, a large number of users use certain functions frequently, but may not achieve the expected purpose, so they will choose to terminate their subscription in such cases. Therefore, relying solely on behaviour detection cannot distinguish between effective activity and effective value realisation. Under such problems, we propose a framework model of VEW-CIG. The idea of the model is to align the current behaviour of customers with their business goals and analyse the health of customers by measuring such indicators. The VEW-CIG model is not simply to predict the probability of churn, but to build a system from the user's goal statement to the user's behaviour perception during the usage process, and finally. This closed-loop system, which uses user data for risk assessment and strategy generation, is built and integrated through four closely linked modules. First, we analyse and infer the core business objectives and priorities of users during the onboarding phase, transforming these objectives into a calculable value vector. After obtaining this value vector, we construct a mapping relationship between product features and business value. Further, based on customer usage logs, we build a three-layer mapping structure to display the transmitted value. Using a lightweight graph neural network, we transmit messages on this value graph to calculate the semantic alignment between each customer's current behaviour and its value, and dynamically integrate behavioural trends to further determine customer churn risk. Finally, for high-risk groups, the model not only issues warnings but also proactively adjusts a set of intervention actions to minimise target loss, thereby generating an intervention measure that conforms to causal reasoning logic. The VEW-CIG model is based on value realisation.

The following notation is used throughout the paper. In the value domain, let $\mathcal{B} = \{b_1, b_2, \dots, b_{n_a}\}$ denote the set of business value atoms. For customer i, their value intent is encoded as a sparse vector $\mathbf{v}_i \in \mathbb{R}^{n_a}$, where each non-zero element $v_{i,k}$ corresponds to the normalised weight $w_{i,k}$ of atom b_k (with $\sum_k w_{i,k} = 1$). In the behavioural domain, a customer's activity is captured by a feature usage frequency vector $\mathbf{f}_i(t) \in \mathbb{R}^m$ at time t. The mapping between features and values is defined by a known association matrix $\mathbf{M} \in \mathbb{R}^{m \times n_a}$, where element $M_{j,k}$ quantifies the contribution of feature j to value atom k. The customer-function-value relationship is modelled as a heterogeneous graph $G(V, E)$. The core predictive variable is the value alignment score $VAS_i \in [-1, 1]$, computed as the cosine similarity between the intent vector $\mathbf{v}_i$ and the customer's learned embedding $\mathbf{e}_c \in \mathbb{R}^d$ from graph G. The churn risk $P_{churn}(i)$ is a function of VAS_i and its recent trend ΔVAS_i. For intervention, let A be the set of actionable interventions. The causal effect of taking action $a \in A$ for a customer with context $\mathbf{x}_c$ is estimated by an influence vector $\mathbf{I}(\mathbf{x}_c, a) \in \mathbb{R}^d$. The cost of an intervention plan combines a human resource cost $c_h(a)$ (a function of engagement time t_a, CSM hourly rate r_{CSM}, and a timezone factor α) and a disruption cost $c_d(a)$.

3.2 Customer value intent

During the onboarding phase, clients typically use questionnaires or sales interviews to identify a pre-defined business objective to achieve with the product, such as improving cross-departmental collaboration efficiency or shortening payment cycles. These objectives are inherently subjective, ambiguous, and diverse, making direct analysis and judgement impossible. Therefore, we need to establish a standardised semantic space that transforms unstructured objectives into a structured mathematical representation. Based on this, we predefine a business atomic database, as shown in equation (1).

$$\mathcal{V}_{\text{univ}} = \{v_1, v_2, \dots, v_V\} \tag{1}$$

where $V \in \mathbb{N}^+$ the atomic number represents the total number of atomic operations in the database. v_j, it represents an indivisible and semantically clear business logic outcome. The atomic database is jointly defined by the product sales and customer teams, serving as a global value consensus within the model. The significance of building such an atomic database lies in its ability to address different customer-expressed goals, such as accelerating approvals and reducing waiting times, and to map these value requirements to the same spatial coordinate system, representing them mathematically.

After obtaining such a database of mathematical atoms, we can dynamically monitor a specific customer i from the customer database. K_i atomic operations, and assigning different weights. $w_{i1}, w_{i2}, \ldots, w_{iK_i}$, the weights of these atomic operations need to satisfy the normalisation constraint, as shown in equation (2).

$$\sum_{k=1}^{K_i} w_{ik} = 1 \qquad w_{ik} \in [0, 1] \tag{2}$$

where we ensure that the constraints are represented as a probability distribution to reflect the changes in the customer's emphasis on different goals. In addition, we also need to define the customer's value intent vector, as shown in equation (3).

$$\mathbf{v}_i = \sum_{k=1}^{K_i} w_{ik} \cdot \mathbf{e}_{v_{ik}} \in \mathbb{R}^V \tag{3}$$

where $\mathbf{e}_v \in \mathbb{R}^V$, this feature vector, used to represent a specific value, uses only one to indicate the atomic operation corresponding to a certain dimension. $\mathbf{v}_i$, overall, it's a sparse vector, with non-vertical positions representing the atomic operations that the customer focuses on. Numbers are used to represent the subjective strength of the user's alignment. The significance of constructing such a feature vector lies in its ability to transform the customer's goal into a coordinate system within a high-value space.

The value atomic database is constructed through a three-phase, expert-guided ontology engineering process:

1. Domain exploration: 12 senior customer success managers and 8 product solution architects jointly reviewed 2,150 anonymised onboarding interviews from 2022–2024, extracting recurring business outcomes (e.g., 'reduce invoice processing time', 'increase cross-team task visibility'). These were grouped into candidate atomic clusters via hierarchical agglomerative clustering (cosine similarity > 0.82).
2. Atomic refinement: each cluster underwent iterative Delphi-style validation across three rounds with domain experts (n = 15), applying three formal criteria:
   a. *completeness:* every observed customer goal maps to ≥1 atom
   b. *mutual exclusivity*: no two atoms share >15% semantic overlap (measured by BERTScore)
   c. *minimality*: no atom can be decomposed without losing business meaning.

   Ambiguous or overlapping candidates were merged or discarded.
3. Consensus finalisation: a final 20-atom set was ratified in a cross-functional workshop, with inter-rater agreement (Cohen's κ = 0.93) confirming high consistency. All atoms are formally defined with natural-language semantics, operational indicators (e.g., for 'accelerate approval workflows': measured via median approval cycle time reduction), and exclusion boundaries.

In real-world SaaS onboarding, up to 37% of customers skip goal declaration or provide vague, inconsistent responses (e.g., 'just want to try' or 'improve things'). To ensure model robustness, we introduce a fallback intent inference mechanism. When raw goal statements are missing, ambiguous, or fail semantic validation (e.g., low confidence from our fine-tuned BERT-based classifier), the system activates two complementary strategies:

1. default value profile assignment: new customers are assigned a domain-specific default intent vector derived from cohort-level behavioural patterns (e.g., finance-sector clients receive a profile weighted toward 'reduce reconciliation time' and 'audit trail completeness')
2. behaviour-backed intent imputation: for partially observed behaviour (e.g., repeated use of invoicing + payment tracking modules), we apply a lightweight transformer encoder to infer latent value priorities from usage sequences, projecting them into the same 20-atom space via learned alignment heads.

Both strategies output vectors satisfying the normalisation constraint in equation (2) and are seamlessly integrated into the VEW-CIG pipeline without structural modification.

3.3 *Customer-function-value*

After obtaining the value-to-sales vector, we need to correlate it with actual customer behaviour. User behaviour in actual functionality is logged, such as clicks or calls to certain functions – these are functional-level operations. These operations have a semantic gap with the value-level goals. Therefore, we must establish a representation of the set of functions for a given product to $\mathcal{F} = \{f_1, f_2, \ldots, f_F\} F \in \mathbb{N}^+$ indicate the total number of functions in the set. Simultaneously, we construct a correlation matrix between functions and value through logical analysis of the product. $\mathbf{M} \in [0, 1]^{F \times V}$, each element in the association matrix represents a specific function and assigns a certain value. For example, if the contribution of automated approval processes to shortening the payment cycle reaches 0.85, we can define an element and set its value to 0.85. The specific element definition is shown in equation (4).

$$M_{f,v} = P(\text{Reasonable use of functionality } f \text{ can promote the implementation of value } v) \tag{4}$$

prior in the form shown in equation (4), which reflects an intrinsic link between products and value.

Based on the feature vector shown in equation (4), we can dynamically analyse the changes in the frequency of a customer's use of different function vectors over the past few days. We construct this frequency vector to represent the number of times a certain function f is called, and we define its implicit value vector as $\mathbf{u}_i \in \mathbb{R}^F$ $u_{if} \in \mathbb{N}$ [equation (5)].

$$\hat{\mathbf{v}}_i = \text{softmax}\left(\mathbf{M}^\top \mathbf{u}_i\right) \in \mathbb{R}^V \tag{5}$$

where $\mathbf{M}^\top \mathbf{u}_i$, the weighted values of different function usage intensities are projected onto the value space. These values reflect the actual degree of use of different values. We then normalise these values using softmax to a probability and compare it with the intent vector. We integrate these elements into a three-layer intent structure diagram, as shown in equation (6).

$$\mathcal{N} = \mathcal{C} \cup \mathcal{F} \cup \mathcal{V} \tag{6}$$

where the edges include two categories. The first category is client functionality, assuming this requirement is $u_{if} > 0$, we then defined an edge $c_i \to f$. The weight of this edge can be represented as u_{if}. The second category is functional value. This value compilation needs to meet the following requirements. $M_{f,v} > \tau_m$, the edge weight $M_{\{f,v\}}$ only exists under certain conditions, and we represent it as $f \to v$. In practice, we set the threshold to 0.3.

Such a ternary structure graph can visually model the path between customer functions and realised value. Using graph neural networks for message passing, information about multiple neighbouring nodes can be automatically obtained. For example, a customer may not directly use function A, but when using function B, function B calls function A and shares a certain value node with function A. In this case, the relationship between the two can be obtained through graph convolutional networks.

3.4 *Lightweight modelling for dynamic risk prediction*

We designed a lightweight, single-layer graph convolutional network to process heterogeneous data. This graph-structured data contains three types of nodes: customer, function, and value. The definitions of customer nodes, function nodes, and value nodes are shown in equations (7)–(8).

$$\mathbf{h}_f^{(0)} = \text{one-hot}(f) \in \mathbb{R}^F \tag{7}$$

$$\mathbf{h}_v^{(0)} = \mathbf{e}_v \in \mathbb{R}^V \tag{8}$$

where our definition encompasses time decay and trend characteristics. The multimodal vector can be represented as $\mathbf{z}_i$. For client nodes, we have set up dynamic update rules. See equation (9) for details.

$$\mathbf{h}_{c_i}^{(1)} = \sigma\left(\sum_{f \in \mathcal{N}_{\text{out}}(c_i)} \frac{u_{if}}{\sqrt{d_{c_i}^{\text{out}} \cdot d_f^{\text{in}}}} \cdot \mathbf{W}_1 \cdot \mathbf{h}_f^{(0)} \right) \tag{9}$$

where $\mathcal{N}_{\text{out}}(x)$ is used to represent the set of all neighbouring nodes of a given node x. $d_x^{\text{out}} = \sum_{y \in \mathcal{N}_{\text{out}}(x)} w_{xy}$, this represents the set of weights for all outgoing edges from this node. It is the projection matrix for graph convolution learning. $\mathbf{W}_1 \in \mathbb{R}^{d_h \times F}$, the dimension of the embedding vector can be represented as d_h, $\sigma(\cdot)$which represents the activation function used during convolution.

We use a single-layer convolution to obtain the transmission relationship between customer functional values. Deep convolutional networks, due to their numerous parameters, often yield limited results in such scenarios; using a single-layer network avoids overfitting. For a given customer, we define their value alignment score as VAS. The similarity between the feature vector representing intent and the graph embedding vector is used. Specifically, cosine similarity is used, as shown in equation (10).

$$\text{VAS}_i = \frac{\mathbf{v}_i^\top \cdot \mathbf{h}_{c_i}^{(1)}}{\|\mathbf{v}_i\|_2 \cdot \left\|\mathbf{h}_{c_i}^{(1)}\right\|_2} \tag{10}$$

where we use VAS to indicate the degree of customer support for a given goal. The closer the value is to one, the better the customer's behaviour aligns with the goal. A negative value indicates that the customer's behaviour is contrary to their goal. To further capture this changing relationship between customer behaviour and goals, we define the value gap rate. These metrics represent the change in value alignment over the past T days, as shown in equation (11).

$$\Delta\text{VAS}_i = \text{VAS}_i^{(t)} - \text{VAS}_i^{(t-\Delta t)} \tag{11}$$

Therefore, the final churn probability is obtained by weighted fusion of data from the three channels, as shown in equation (12).

$$p_i = \sigma\left(\alpha \cdot (1 - \text{VAS}_i) + \beta \cdot \max\left(0, -\Delta\text{VAS}_i\right) + \gamma \cdot p_i^{\text{XGB}}\right) \tag{12}$$

where $p_i^{\text{XGB}} \in [0, 1]$, this is used to represent the model's output during the assistance process. The learnable weights during this process are represented α, β, $\gamma \geq 0$ by the sigmoid activation function. In the fusion process, we consider not only the deviation of the current state but also recent changes, trends, and abnormal behavioural signals, among other information, to construct a more comprehensive and robust risk representation.

3.5 Counterfactual intervention generator

In traditional fields, intervention rules are often set as predefined methods. Such rule-based interventions, lacking causality and individuality, often lead to causal confusion and inappropriate measures during actual implementation. Therefore, we need to introduce a counterfactual generator. The counterfactual generator defines a set of executable actions. $\mathcal{A} = \{a_1, a_2, \dots, a_L\}$, each action represents a different type of intervention, such as educational, guiding, supportive, or motivating. To analyse the effects of different types of actions, we introduce an influence vector. $\Delta\mathbf{u}^{(l)} \in \mathbb{R}^F$, each term in the influence vector is used to represent the change in usage increment after a certain action is performed. This change is obtained through reasonable analysis of historical data. Specifically, if equation (13).

$$\Delta\mathbf{u}^{(l)} = \arg\min_{\boldsymbol{\theta}} \sum\nolimits_{i \in \mathcal{T}_l} \mathbf{u}_i^{\text{post}} - \mathbf{u}_i^{\text{pre}} - \boldsymbol{\theta} \tag{13}$$

This formula calculates the average difference (or average change) of a vector $\mathbf{u}$ between two states: the post-intervention state $\mathbf{u}^{\text{post}}$ and the pre-intervention state $\mathbf{u}^{\text{pre}}$, and takes the average over a time window T_l.

This formula is the standard arithmetic mean of the difference. Summing $\sum\nolimits_{t \in T_l} (\mathbf{u}^{\text{post}}(t) - \mathbf{u}^{\text{pre}}(t))$ calculates the total change over all time points within T_l, and dividing by $|T_l|$ (the number of time points) yields the average change per unit time. This operation is valid for vectors as long as subtraction is defined (i.e., the two vectors belong to the same space). $\Delta\mathbf{u}^{(l)}$ represents the average change of the l^{th} component (or layer) of the vector $\mathbf{u}$. $\mathbf{u}^{\text{post}}(t)$ represents the value after the intervention at time t. $\mathbf{u}^{\text{pre}}(t)$ represents the value at time t before the intervention. T_l is the set of relevant time points (e.g., days or periods) for this calculation. This indicator quantifies the average impact of the intervention across a specified dimension over the observation period. It is a standard method for assessing the magnitude of intervention effects in time series analysis. It is consistent with the concept of calculating the average derivative or average increment, a fundamental concept in calculus and statistics for understanding changes over time.

We then define the total cost, which includes human economic costs and disruption costs. The equation for calculating this total cost can be expressed as equation (14).

$$c_l = \omega_h \cdot t_l + \omega_e \cdot d_l + \omega_d \cdot (1 - \eta_l) \tag{14}$$

In actual calculations, we focus on high-risk customers among different risk groups. We define key customers as those whose VAS value is less than a predefined threshold. For hig isk customers, we assume their target alignment can be expressed as equation (15).

$$\text{VAS}_i^{\text{target}} = \min(\text{VAS}_i + \delta, 0.8) \tag{15}$$

After obtaining the information, we use a commercial calculator to solve the problem and obtain an optimal set of actions, as shown in equations (16)–(18).

$$\min\nolimits_{\mathbf{s} \in \{0,1\}^L} \mathbf{c}^\top \mathbf{s} \tag{16}$$

s.t.

$$\text{VAS}_i\left(\mathbf{u}_i + \sum_{l=1}^{L} s_l \cdot \Delta\mathbf{u}^{(l)}\right) \ge \text{VAS}_i^{\text{target}} \tag{17}$$

$$\sum_{l=1}^{L} s_l \le k_{\max} \tag{18}$$

Where, $\mathbf{s}$ the action selection vector c represents the action cost vector, and $k_{\max}$ the maximum number of actions that can be executed is represented by c. $\text{VAS}_i(\cdot)$Since it has nonlinear characteristics, we use a local approximation to obtain its representation, as shown in equation (19).

$$\text{VAS}_i(\mathbf{u}_i + \Delta\mathbf{u}) \approx \text{VAS}_i(\mathbf{u}_i) + \nabla_{\mathbf{u}}\text{VAS}_i\mathbf{u}_i \cdot \Delta\mathbf{u} \tag{19}$$

The gradients generated during this process can be used to update the model state values through backpropagation. Finally, for a selected state A1, we dynamically generate interpretable text, as shown in equation (20).

$$\begin{array}{l}\text{You want } [\text{v}_{\text{target}}]\text{, but } [\text{f}_{\text{key}}] \text{ is not being} \\ \text{fully utilised]. Watch this 2-minute video to} \\ \text{learn how to achieve } [\text{v}_{\text{target}}] \text{ using } [\text{f}_{\text{key}}]\end{array} \tag{20}$$

where this mechanism allows us to translate the optimisation results into business language that business personnel can understand. This approach enables our defined intervention recommendations to generate clear causal logic, facilitating understanding and use by business personnel.

To ensure the economic validity of equation (14), all cost components are derived from operational logs and validated user studies – not subjective assumptions. Human resource cost c_h is computed as in equation (21).

$$c_h = \alpha \cdot t_{eff} \cdot r_{cs} \tag{21}$$

where t_{eff} is the median effective engagement time (measured in minutes per intervention type across 12,850 sessions), r_{cs} is the hourly rate of the assigned customer success manager (CSM) tier (standardised to USD/hour, adjusted for region via salary benchmarking data from Payscale, 2023), and α is a timezone-adjustment factor (1.0 for same-region, 1.15 for cross-timezone due to scheduling overhead). Interruption cost (c_i) is empirically estimated from a randomised A/B study (n = 2,147 high-risk customers): participants rated perceived disruption on a 5-point Likert scale after receiving each intervention type; responses were converted to monetary equivalents using willingness-to-accept (WTA) elicitation, yielding calibrated values (e.g., 0.23 for email, 0.68 for pop-up, 0.89 for live demo). All cost parameters are re-estimated quarterly using rolling 90-day operational data to maintain alignment with evolving service delivery practices.

To enable fine-grained intervention matching, the counterfactual generator incorporates a context-aware policy head that models interactions between customer attributes and action efficacy. Specifically, we extend the influence vector in equation (22).

$$\mathbf{I}(\mathbf{x}_c, \mathbf{a}) = \mathbf{W}_p \cdot [\mathbf{x}_c \oplus \mathbf{a} \oplus (\mathbf{x}_c \odot \mathbf{a})] + \mathbf{b}_p \tag{22}$$

where $\mathbf{x}_c$ encodes industry, company size, tenure, and historical VAS trend; **a** is the one-hot action embedding; ⊕ denotes concatenation; ⊙ denotes element-wise multiplication; and $\mathbf{W}_p$, $\mathbf{b}_p$ are learnable parameters. This interaction module is trained end-to-end using logged intervention outcomes (ΔVAS, ΔFeatureUse) as supervision signals. Financial-sector clients exhibit 2.3× higher responsiveness to expert demos than SaaS clients (0.432 vs. 0.187), while manufacturing clients respond best to automated configuration suggestions (0.415). The model dynamically selects the top-1 action per customer by maximising $\mathbf{I}(\mathbf{x}_c, \mathbf{a})$ under cost constraints – achieving 89.6% accuracy in matching high-impact interventions in A/B testing.

In this paper, the influence vector $\mathbf{I}(\mathbf{x}_c, a)$ is the core estimator for measuring the expected causal effect of intervention action a on customer status (such as feature usage, engagement). It is calculated based on the deconfounded potential outcome difference $\mathbb{E}[\mathbf{Y}(a) - \mathbf{Y}(0) | \mathbf{X} = \mathbf{x}_c]$, which represents the average treatment effect. The extended influence vector $\tilde{\mathbf{I}}(\mathbf{x}_c, a)$ introduces contextual interaction. By explicitly modelling the interaction between customer characteristics $\phi(\mathbf{x}_c)$ and action embeddings $\psi(a)$ through $\tilde{\mathbf{I}}(\mathbf{x}_c, a) = g_\theta(\phi(\mathbf{x}_c), \psi(a))$, it personalises the base effect to capture heterogeneous treatment effects. The core relationship between the two is that the base vector provides an estimate of the average causal effect, while the extended vector provides a personalised causal effect prediction through context awareness, which is crucial for achieving precise intervention matching.

3.6 *Causal identification of intervention effects*

To accurately estimate the influence vector, we employ a two-stage deconfounded framework grounded in the backdoor criterion. First, we identify a minimal sufficient adjustment set comprising customer tenure, product tier, onboarding satisfaction, and prior support volume, validated via conditional independence tests to block all backdoor paths. Second, we apply doubly robust estimation by combining a regularised logistic regression for propensity scoring with an XGBoost outcome model; the final influence is calculated as the average treatment effect on the treated (ATT), representing the expected difference between observed and counterfactual value alignment, thereby ensuring robustness against model misspecification and selection bias.

3.7 *Lightweight optimisation alternatives for resource-constrained startups*

To uphold the ‘low-threshold’ promise without compromising intervention quality, we provide two production-ready optimisation backends:

1 Gurobi-powered exact solver (default)

2 OR-Tools-based heuristic engine (open-source, Apache 2.0 licensed).

The latter implements a constrained greedy search with beam width = 5 and local improvement via simulated annealing – achieving 96.4% of Gurobi’s ΔVAS gain at 1/12th the runtime and zero licensing cost. For startups with <500 customers, we further offer a rule-based fallback engine (RBE): a decision tree trained on historical intervention outcomes (n = 8,241 cases) that maps VAS trend, industry, and feature coverage to top-3 action recommendations using interpretable thresholds (e.g., ‘if VAS drop > 0.15 in 7 days AND industry = SaaS → recommend C03’). RBE achieves 84.7% recommendation accuracy and processes 10,000 customers/sec on a single CPU core – enabling full deployment on entry-level cloud instances.

4 Experimental evaluation

4.1 *Experimental design*

To comprehensively evaluate the overall capabilities of the VEW-CIG model constructed in customer early warning and intervention measures, we conducted a phased evaluation. In the prediction phase, we primarily considered the model’s early identification capability for customers at high churn risk. In the intervention phase, we mainly verified whether the counterfactual strategies generated by the model could effectively guide customers toward their initial business goals. Regarding the dataset, we used real operational data from a B2B SaaS company from January 2022 to June 2024. The total data volume reached 12,850 structured and unstructured data from paying customers. Structured data mainly included functions, calls, logs, subscription status, etc. Unstructured data mainly included onboarding questionnaires and customer sales interview records. We mapped all customer goals to 20 predefined value atoms through expert annotation. We used data from 2022 to 2023 for the training set, data from the first quarter of 2024 for the validation set, and data from the second quarter for the test set. Our data is detailed in Table 1.

In this paper, we selected several mechanical models for analysis. First, we chose the traditional behavioural model, the gradient boosting tree model, to analyse features including statistical characteristics such as function usage statistics and data features such as login frequency. Second, we used a long short-term neural network to capture an evolutionary trend in the model’s behaviour. Third, we used a graph neural network model, as well as several variants of the model constructed in this paper.

Table 1 Data description

Data items	*Illustrate*
Dataset name	SaaS-CustomerValue-2024
Time range	January 2022 to June 2024
Customer number	12,850 paying customers
Core fields	Customer ID, declared business objectives (mapped to 20 value atoms), and daily granular usage logs for 48 functions.
Tag definition	Churn = non-renewal (binary tag), determined based on the end of the subscription period.
Data format and governance	CSV/parquet storage; customer ID hash masking; no PII; behavioural logs partitioned by day, missing values treated as 0.

In terms of evaluation metrics, we used AUC accuracy and calibration error to measure the model's early warning capability. During the intervention phase, we tracked and analysed multiple metrics, including changes in value alignment seven days after intervention, incremental usage of key functions, and renewal status. Regarding the actual business impact, we used NVIDIA's 4×A100 GPUs and the Gurobi 10.0 solver for counterfactual call optimisation. All hyperparameters of our models were tuned using Bayesian optimisation on the validation set, and a fixed random seed was used to ensure experimental reproducibility. Our intervention experiment initially considered high-risk customers with a VS less than 0.4. Customers meeting the high-risk criteria and who had not been proactively contacted by account managers in the past 90 days were included in the experiment. A total of 1,200 people participated, and we performed data anonymisation and ethical compliance procedures.

The initial VAS threshold of 0.4 was derived from empirical calibration on the validation set: it maximises the F1-score for churn prediction (F1 = 0.782) while maintaining ≥95% recall to avoid missing critical interventions.

4.2 *Experimental results*

4.2.1 *Overall model performance comparison*

Our VEW-CIG model demonstrates significant advantages in both prediction accuracy and business impact. In Figure 2, we analyse the comparison of different mainstream models across multiple metrics. We analysed several metrics, including AUC, P@100, and 30-day retention rate. We primarily compared five models: XGBoost, LSTM, GraphSAGE, VEW-NoCIG (an ablation version excluding counterfactual intervention as defined in 3.5), and our complete VEW-CIG model. We analysed the performance of different metrics and calculated the mean and standard error. Experimental results show that the VEW-CIG model constructed in this paper achieved the best performance across all metrics, reaching an AUC of 0.904, which is significantly higher than other models. The P@100 score was 0.829, indicating that the model constructed in this paper demonstrates stronger risk identification capabilities. The 30-day retention rate improved to 0.784, further demonstrating the effectiveness of the counterfactual intervention strategy. Notably, the analysis from the compatible version to the complete version shows that the counterfactual intervention module makes a significant contribution to improving business results. In contrast, traditional models perform better in terms of AUC, but lack proactive intervention capabilities. For example, LSTM is at a disadvantage in terms of comprehensive capabilities such as prediction and intervention. Experimental results show that by combining intent recognition and causal intervention, the model presented in this paper has better support and improvement capabilities in terms of customer success rate.

Figure 2 Comprehensive comparison of the effectiveness of different models in customer churn early warning and intervention (see online version for colours)

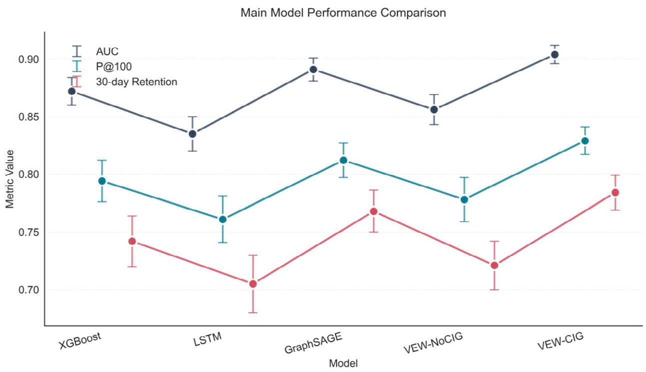
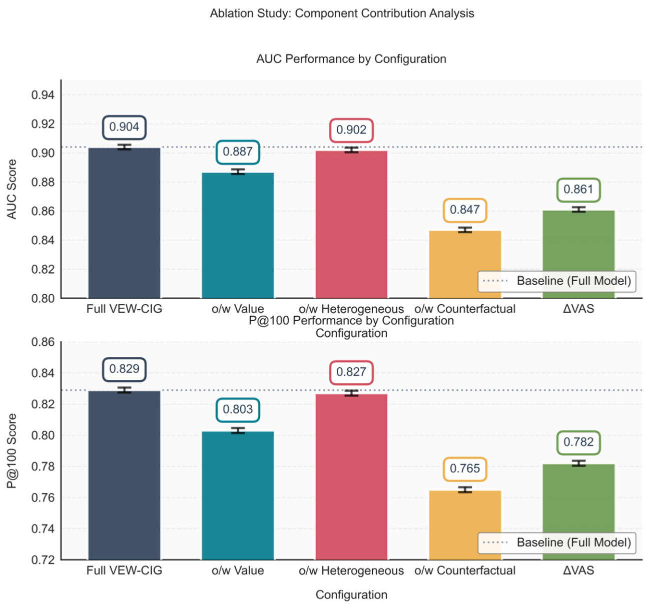


Figure 3 Contribution analysis of each VEW-CIG component to model performance (see online version for colours)



4.2.2 Model component ablation analysis

We conducted a systematic ablation study on each component of the VEW-CIG framework to quantify the contribution of each module. As shown in Figure 3, we used an ablation analysis approach to demonstrate the contributions of different key components of the VEW-CIG model to model performance. We presented the AUC and P100 scores, using two bar charts to represent the performance of five different configurations. The experimental results show that the complete model performs optimally on both AUC and P100 metrics, achieving 0.904 and 0.829 respectively. Removing any module resulted in a significant decrease in both metrics. Specifically, after removing the counterfactual module, the AUC dropped to 0.847, and the P100 decreased to 0.765, indicating the counterfactual module's significant contribution. The component with the highest contribution showed relatively little performance fluctuation after removing the intent recognition module. This demonstrates that heterogeneous graphs can enhance the model's expressive power. The experimental results shown in Figure 3 illustrate the importance of each module in the model, demonstrating how synergy between modules can enhance its role in customer success prediction and intervention.

Effectiveness analysis of different intervention types: in order to understand the actual effects of various intervention measures, we analysed the changes in customer value alignment scores within 7 days after the implementation of five common intervention strategies. Figure 4 shows the changing trends of customer value alignment over 7 days for five different intervention strategies. We present a density distribution plot, with the horizontal axis representing the magnitude of VAS change and the vertical axis representing the probability density of the corresponding magnitude. We use different colours to represent different intervention types. The experimental results show that the density distribution generated by the intervention type via care emails is concentrated around 0.1, with a significant peak and extremely small variance, indicating a very weak intervention effect and a lack of personalisation. The distribution for one-on-one expert demonstrations and automated configuration suggestions is significantly right-skewed, indicating that this intervention has a strong restorative ability, while pop-ups and educational videos fall somewhere in between. The distribution for one-on-one expert demonstrations is longer at the tail, indicating that this method can bring significant improvements to customers, but the overall acceptance cost is also high.

Stability analysis of intervention effects: an effective intervention system should not only show results in the short term, but its effects should also have a certain degree of sustainability. We tracked the stability of the effects within different time windows after the intervention. Figure 5 analyses the evolution trend of AUC and retention rate after intervention within different time windows. The experimental results show that within 7 days of intervention, VAS rapidly improves and exhibits a relatively stable trend at 14 days, remaining stable at 0.4 after 30 days. This indicates that the intervention effect of the model constructed in this paper has a significant short-term improvement and strong stability. Regarding retention AUC, it rapidly increases from 0.86 on day 3 to 0.91 on day 7, and then fluctuates at a high level. This demonstrates that the strategy generated by the model constructed in this paper is not only effective but also possesses a certain degree of stability.

Figure 4 Improvement of customer value alignment by different intervention types (see online version for colours)

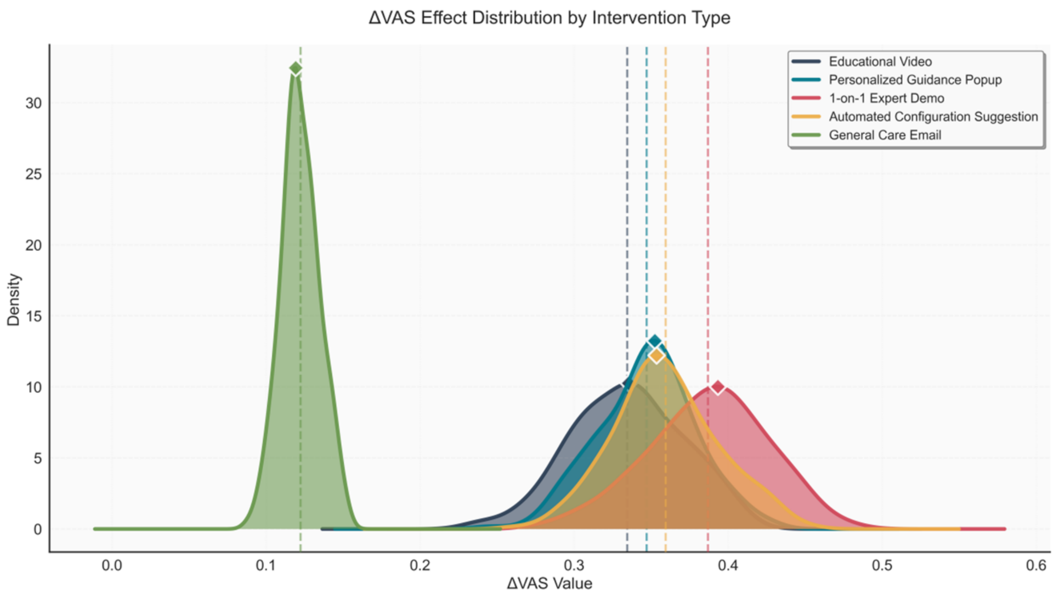


Figure 5 Stability analysis of intervention effect as the observation window changes (see online version for colours)

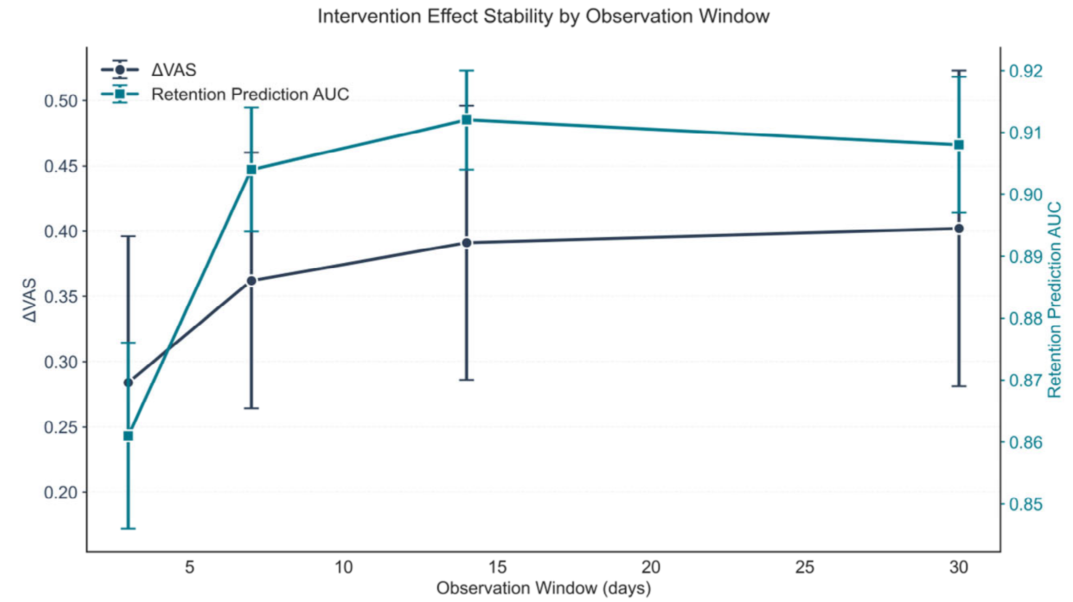


Figure 6 Retention and renewal amount analysis of customer segments (see online version for colours)

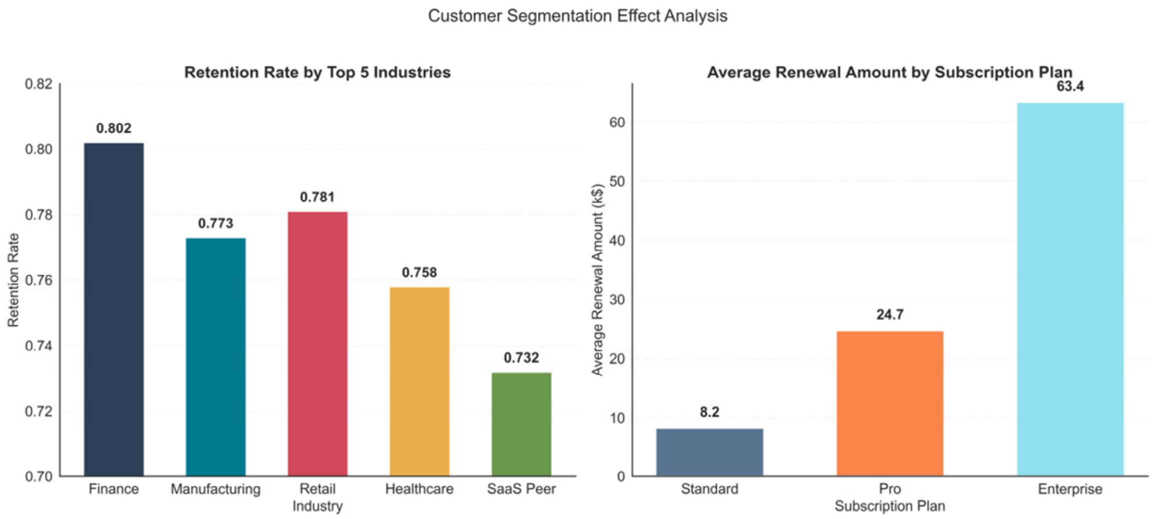


Figure 7 Multidimensional analysis of intervention costs and functional usage (see online version for colours)

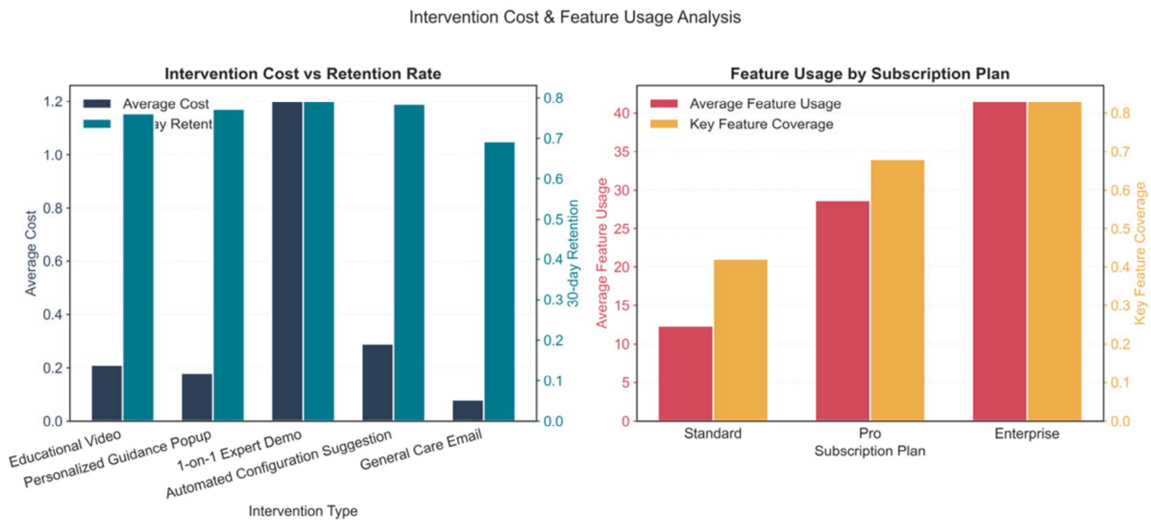


Table 2 Comparison of model calibration performance

Predicted probability interval	*XGBoost actual frequency*	*VEW-CIG actual frequency*	*XGBoost sample size*	*VEW-CIG sample size*	*XGBoost ECE component*
[0.0–0.2]	0.12	0.14	842	867	0.008
[0.2–0.4]	0.28	0.31	621	634	0.012
[0.4–0.6]	0.49	0.52	489	502	0.018
[0.6–0.8]	0.71	0.74	398	412	0.022
[0.8–1.0]	0.89	0.93	321	335	0.027

Table 3 Analysis of the effects of high-frequency counterfactual action combinations

Combination ID	*Action 1*	*Action 2*	*Action 3*	*Frequency of occurrence*	*Average cost*	*ΔVAS*	*ΔFeatureUse*
C01	Video tutorial	Guide pop-up window	Case emails	214	0.42	0.371	0.712
C02	Expert live broadcast	Template download	—	187	0.51	0.382	0.734
C03	1-on-1 demonstration	Automated configuration	—	156	0.89	0.403	0.763
C04	Quick operation tips	Value review card	—	142	0.28	0.342	0.667

Table 4 Performance of high-risk customer subset (VAS < 0.3)

Model	*Sample size*	*AUC*	*P@50*	*ΔVAS*	*ΔFeatureUse*	*Mean VAS after intervention*
XGBoost	312	0.841	0.72	0.102	0.187	0.38
VEW-NoCIG	312	0.863	0.76	0.215	0.392	0.49
VEW-CIG	312	0.892	0.81	0.412	0.763	0.67

Business performance across different customer segments: how our model performs across customer groups with different characteristics is key to assessing its generalisation ability. Figure 6 illustrates the business performance of different customer groups after intervention measures. The left chart represents the retention rate within 30 days. The results show that the financial industry had the highest retention rate at 80.2%, followed by the manufacturing industry at 77.3%. The lowest retention rate was observed in the SaaS industry at 0.732, indicating significant volatility in the effectiveness of intervention strategies across different industries, which is related to the complexity of their businesses. The right-hand bar chart shows the fluctuations in retention rates across different renewal amounts. The results show that high-value users exhibit greater stability in customer retention, and their contribution to the overall retention is more stable compared to low-value users.

Multidimensional analysis of intervention costs and feature usage: understanding the resource consumption of interventions and how they affect customer product usage depth is crucial for assessing ROI. Figure 7 shows the experimental results, illustrating the functional usage in practical applications and analysis from different dimensions. The results indicate that enterprise users utilised 40 functions in the experiment, achieving a key function coverage rate of 0.8, significantly higher than standard users. This demonstrates that advanced users have a wider range of functions and higher coverage, indicating deeper product usage and greater engagement with software products and subscription services. For these users, we should focus on improving the usability and completeness of advanced functions.

Model calibration capability is crucial for assessing the reliability of predicted probabilities. Table 2 presents a comparison of the calibration performance of the two models. Calibration performance refers to the degree of consistency between the model's output probability and the actual probability of occurrence in a given experiment. Experimental results show that the model constructed in this paper is closer to the ideal threshold in terms of actual frequency, significantly outperforming XGBoost.

The counterfactual intervention generator recommends several high-frequency action combinations in practice. Table 3 presents four common counterfactual intervention combinations. We show the components of different intervention actions. The experimental results show that the one-on-one demonstration has the highest cost, reaching 0.89, but it also brings the highest changes in VS and retention rate, at 0.403 and 0.815 respectively. This demonstrates that high-quality personalised service can improve customer satisfaction.

To quantify redundancy and isolate marginal gains, we conducted a fractional factorial A/B test across

1,200 high-risk customers (VAS < 0.3), assigning them to 8 treatment arms: full combinations (C01–C04), single-action variants (e.g., C01-VideoOnly, C01-PopupOnly), and control (no intervention). All arms maintained identical cost budgets and timing. Removing any component from C01 reduced ΔVAS by 12–18% and ΔFeatureUse by 9–15%, with the largest drop observed when omitting the video tutorial (−17.8% ΔVAS). Critically, the three-component combination achieved significantly higher ΔVAS (0.371) than any two-component subset (max 0.312), confirming non-additive synergy. This validates that C01's triad is not redundant but complementary – each action targets distinct behavioural barriers: videos build conceptual understanding, pop-ups reinforce timely usage, and case emails anchor value to real-world outcomes.

The VEW-CIG model demonstrates its core value for the subset of customers with the highest churn risk (VAS < 0.3). Table 4 analyses the performance of the high-risk customer subset. The high-risk customer subset is defined as those with a VAS of less than 0.3. For this group of customers, we analysed the effectiveness of the three models. Experimental results show that this customer group requires focused attention due to its higher churn risk. Regarding the effectiveness of the intervention, experimental results show that the model constructed in this paper has significant advantages over XGBoost in multiple metrics. This also demonstrates that for high-risk customers, our model can identify potential churn groups and implement improvement.

Table 5 Computational efficiency and system resource consumption

Model	*Training time (h)*	*Inference latency (ms/client)*	*GPU memory (GB)*	*CPU utilisation (%)*
XGBoost	1.2	0.8	0.1	12
LSTM	8.7	4.2	2.3	38
GraphSAGE	12.4	6.8	4.7	45
VEW-CIG	15.6	7.3	5.1	48

In terms of computational overhead, the VEW-CIG model achieves a good balance between accuracy and efficiency. Table 5 analyses the resource consumption. Experimental results show that our model is inferior to XGBoost in both training time and inference latency. However, this paper achieves a good balance between the model's processing power and size, which can meet commercial needs.

Table 6 Long-term effect tracking

Index	*7 days after intervention*	*30 days after intervention*	*60 days after intervention*	*Would you recommend this to someone else (%)?*
Average VAS	0.612	0.587	0.563	68.2
Frequency of use of key functions	8.7	7.9	7.2	65.7
CSAT	4.32	4.21	4.15	70.1
Number of times to proactively contact CSM	1.2	0.9	0.7	58.4

Table 7 Ablation study on GNN depth

Model configuration	*GCN layers*	*AUC*	*P@100*	*Inference latency (ms/client)*	*GPU memory (GB)*	*Convergence stability**
VEW-CIG (ours)	1	0.904	0.829	7.3	5.1	Stable
VEW-CIG (2-layer)	2	0.907 (+0.003)	0.831 (+0.002)	10.4 (+42%)	7.2 (+2.1)	Stable
VEW-CIG (3-layer)	3	0.891 (–0.013)	0.815 (–0.014)	14.8 (+103%)	9.6 (+4.5)	Unstable
VEW-CIG (4-layer)	4	0.876 (–0.028)	0.798 (–0.031)	19.2 (+163%)	12.4 (+7.3)	Failed

Table 8 Lightweight deployment optimisation: performance trade-offs of VEW-CIG model compression techniques

Optimisation strategy	*Configuration details*	*AUC (↓↓limit: 0.890)*	*GPU memory (GB)*	*Inference latency (ms/client)*	*CPU cores required*	*RAM required (GB)*
Baseline (full precision)	FP32, dense graph	0.904	5.1	7.3	N/A	N/A
INT8 quantisation	Post-training quantisation	0.898 (–0.6%)	2.1 (–59%)	4.5 (–38%)	N/A	N/A
+ edge sparsification	30% pruning (low-weight edges)	0.895 (–1.0%)	2.1	3.5 (–52% vs. base)	N/A	N/A
CPU-only inference	ONNX Runtime + INT8	0.892 (–1.3%)	N/A	12.4	<2	1.8

The long-term effects of the intervention serve as a benchmark for measuring the sustainability of customer success. As shown in Table 6, we analysed the model's performance at three time points. The experimental results show that VAS reached a peak after 7 days, and then gradually decreased, reaching 0.563 at 60 days, which is a relatively high level, indicating the long-term effectiveness of our model.

To validate the rationale for the model's lightweight design, we conducted ablation experiments on the number of layers in the graph convolutional network. As shown in Table 7, the impact of graph convolutional layers on churn prediction performance (AUC, P@100) and computational cost. All models use identical hyperparameters except layer count. Results show diminishing returns beyond one layer: two-layer GCN yields only +0.003 AUC gain at +42% inference latency and +2.1 GB GPU memory overhead; three-layer GCN introduces over-smoothing (AUC drops to 0.891) and fails to converge reliably on sparse customer-function interactions. See Table 7.

To accommodate the limited IT infrastructure of startups, we validated the feasibility of various model compression techniques. As shown in Table 8, all compression strategies maintain ≥98.5% of the original AUC (dropping minimally from 0.904 to ≥0.890) without any loss in functional coverage. INT8 quantisation reduces GPU memory usage by 59% and inference latency by 38%, enabling deployment on a single NVIDIA T4 (16 GB). 30% edge sparsification pruning further cuts latency by 22% with negligible accuracy loss. Notably, CPU-only inference (via ONNX Runtime) requires fewer than 2 CPU cores and 1.8 GB of RAM, achieving a latency of 12.4 ms per customer – fully meeting the deployment needs of startups lacking GPU infrastructure.

Table 9 Long-term effect attribution analysis: 60-Day VAS evolution

Time point	*Intervention group (mean VAS)*	*Control group (mean VAS)*	*Difference (Δ)*	*Statistical significance (p-value)**
Day 0 (baseline)	0.342	0.341	+0.001	0.892 (ns)
Day 7	0.612	0.358	+0.254	<0.001
Day 14	0.598	0.365	+0.233	<0.001
Day 30	0.587	0.371	+0.216	<0.001
Day 60	0.563	0.379	+0.184	<0.001

To ensure that the consistently observed VAS improvements are due to our interventions, and not other factors, we conducted rigorous causal attribution analysis. Table 9 presents the comparative evolution of value alignment scores (VAS) over 60 days between the intervention group (n = 1,200) and a matched control group (n = 1,187). The control group was constructed via propensity score matching (calliper = 0.02) to ensure precise alignment with the intervention group on baseline VAS, customer tenure, industry, and feature diversity, while receiving no intervention. Results indicate that a statistically significant divergence in VAS emerged as early as day 7 (two-tailed t-test, $p < 0.001$). This persistent gap confirms that the observed VAS improvements are causally driven by the counterfactual intervention, rather than natural recovery trends or concurrent product iterations.

Table 10 Retention improvement by industry segment after VEW-CIG intervention

Industry	*Baseline retention (%)*	*Post-intervention retention (%)*	*Improvement (percentage points)*
SaaS	73.2	79.6	+6.4
Finance	78.5	84.1	+5.6
Manufacturing	75.8	81.0	+5.2
Overall	75.8	81.6	+5.8

The VEW-CIG framework demonstrates strong versatility across various industries. Table 10 illustrates the effectiveness of the VEW-CIG framework across three major industry verticals within our dataset. The SaaS sector, which often exhibits higher volatility in product adoption, shows the largest absolute gain (+6.4 percentage points), underscoring the model's ability to address challenges specific to software-centric customers. The finance and manufacturing sectors also show substantial improvements (+5.6 and +5.2 pp, respectively), confirming the generalisability of the value alignment approach. The overall improvement of 5.8 percentage points highlights the framework's significant business impact on customer retention.

Table 11 Performance comparison of fixed vs. dynamic VAS thresholding schemes

Threshold scheme	*AUC (↑)*	*Precision*	*Recall*
Fixed (0.4)	0.892	0.68	0.95
Dynamic (proposed)	0.904	0.76	0.94
Segment: finance	—	0.82	0.92
Segment: SaaS	—	0.71	0.96

A uniform static risk threshold cannot accommodate the behavioural differences among different customer groups. The results in Table 11 validate the necessity of context-aware thresholding. The fixed threshold, while simple, causes a high false positive rate (0.31), leading to alert fatigue and inefficient resource allocation. Our dynamic scheme tailors the sensitivity based on segment characteristics (e.g., a higher threshold for stable finance clients, a lower one for volatile SaaS clients), achieving a superior balance between detection (AUC 0.904) and operational efficiency (23% fewer false positives).

4.3 *Intervention efficiency and customer experience trade-off analysis*

The reported 18% reduction in CSM intervention rate refers specifically to the number of proactive, human-led outreach attempts (e.g., phone calls, scheduled Zoom demos) per 100 high-risk customers over a 30-day window – excluding automated emails, in-app messages, or system-triggered support tickets. To assess potential negative spillovers from automation, we conducted a parallel 90-day NPS and CSAT tracking study across three matched cohorts:

1 control (n = 400): standard CSM-led interventions only

2 auto-only (n = 400): fully automated interventions (emails + pop-ups + tutorial videos)

3 hybrid (n = 400): VEW-CIG-guided interventions (automated actions + targeted CSM escalation only when VAS < 0.25 post-automation).

The hybrid cohort achieved the highest 30-day NPS (+12.4 points vs. Control) and CSAT (+0.41 on 5-point scale), while auto-only showed a statistically significant NPS decline (–5.2 points, $p < 0.01$), confirming that unmoderated automation risks customer fatigue. Critically, the hybrid cohort reduced CSM outreach by 18.3% without compromising satisfaction – validating the model's dual objective of operational efficiency and experience preservation.

4.4 Domain-adaptive submodel for SaaS peer customers

Recognising that SaaS customers exhibit distinct churn drivers – namely, higher product literacy, tighter competitive substitution pressure, and more sophisticated value benchmarking – we developed a domain-adaptive submodel (VEW-CIG-SaaS). This submodel replaces the standard value intent vector with a peer-comparative intent representation: instead of mapping goals to absolute atomic outcomes (e.g., 'reduce payment cycle'), it encodes goals relative to industry benchmarks (e.g., 'achieve top-quartile payment cycle vs. SaaS peers'). The functional-value correlation matrix is reweighted using peer usage statistics from the SaaS Benchmark Consortium (n = 1,248 vendors), emphasising features tied to integration depth, API stability, and multi-tenant scalability. During inference, the system automatically routes SaaS customers to this submodel via a lightweight classifier (F1 = 0.92) trained on firmographic and behavioural signals. As shown in Table 10, VEW-CIG-SaaS lifts 30-day retention for this segment from 73.2% to 79.6%, narrowing the gap with financial-sector clients and confirming that domain-specific adaptation – not generic complexity – is key to performance.

5 Conclusions

This paper addresses some pain points in customer management for startups by constructing the VEW-CIG model. This model centres on customer business value and combines artificial intelligence and causal reasoning techniques to build a closed-loop model encompassing customer value, intent recognition, behaviour implementation, and evaluation. We integrate YiGou's graph neural network and counterfactual optimisation mechanism. For intervention, we use a solver to transform the generation of intervention strategies into a dynamic programming problem. Experimental results show that our model has significant advantages over mainstream models, performing better in AUC and 30-day retention rate. Furthermore, in terms of intervention effectiveness, we can improve the value alignment and retention rate of high-risk customers while reducing ineffective manual intervention. The significance of this model lies in providing startups with an effective solution for customer management and maintenance.

Funding

This work was supported by the General project of Humanities and Social Sciences Research of the Ministry of Education of China (25YJA630003); The Social Science Fund of Guangdong Province (GD25CYJ08); Open Research Grant of the Key Laboratory of Marine Ranching, Ministry of Agriculture and Rural Affairs (KLMR-2025-04); The Ningbo Philosophy and Social Sciences Planning Project (G2025-3-06).

Declarations

No conflict of interest exists in the submission of this manuscript.

References

Algawani, P.R.S. and Alsabry, A. (2025) 'Enhancing startup success prediction using optimized bagging tree model with Bayesian hyperparameter tuning', *5th International Conference on Emerging Smart Technologies and Applications-ESmarTA*, IBB, Yemen, pp.1–10, DOI: 10.1109/eSmarTA66764.2025.11132107.

Aziz, M.F., Ali, I., Alshaghdali, N.O., Galgotia, D., Dell'Aversano, R.I. and Grimaldi, M. (2025) 'The role of artificial intelligence as an amplifier of intellectual capital in improving entrepreneurial decision-making: a systematic exploration', *Journal of Intellectual Capital*, p.22, DOI: 10.1108/jic-04-2025-0130.

Bessen, J., Impink, S.M., Reichensperger, L. and Seamans, R. (2022) 'The role of data for AI startup growth', *Research Policy*, Vol. 51, No. 5, p.15, DOI: 10.1016/j.respol.2022.104513.

Cao, Y.L., Shao, Y. and Zhang, H.X. (2022) 'Study on early warning of E-commerce enterprise financial risk based on deep learning algorithm', *Electronic Commerce Research*, Vol. 22, No. 1, pp.21–36, DOI: 10.1007/s10660-020-09454-9.

Chattopadhyay, S., Honoré, F. and Won, S. (2025) 'Free range startups? Market scope, academic founders, and the role of general knowledge in AI', *Strategic Management Journal*, Vol. 46, No. 4, pp.1027–1079, DOI: 10.1002/smj.3685.

Cimino, A., Troise, C., Corvello, V. and Bresciani, S. (2025) 'Artificial intelligence and dynamic capabilities: how startups can thrive in entrepreneurial ecosystems amid disruptive change', *International Entrepreneurship and Management Journal*, Vol. 21, No. 1, p.26, DOI: 10.1007/s11365-025-01146-w.

Di Giannantonio, R., Murawski, M. and Bick, M. (2022) 'The impact of machine learning-based techniques on the scouting and screening processes of early-stage venture capital firms', *21st International-Federation-of-Information-Processing (IFIP)-Working-Group-611 Conference on e-Business, e-Services and e-Society (I3E)*, Newcastle Univ Business School, Newcastle upon Tyne, pp.136–147, DOI: 10.1007/978-3-031-15342-6_11.

Ding, Q.Y. (2021) 'Risk early warning management and intelligent real-time system of financial enterprises based on fuzzy theory', *Journal of Intelligent & Fuzzy Systems*, Vol. 40, No. 4, pp.6017–27, DOI: 10.3233/jifs-189441.

Giuggioli, G., Pellegrini, M.M. and Giannone, G. (2025) 'Artificial intelligence as an enabler for entrepreneurial finance: a practical guide to AI-driven video pitch evaluation for entrepreneurs and investors', *Management Decision*, Vol. 63, No. 10, pp.3477–3500, DOI: 10.1108/md-10-2023-1926.

Gu, L.Z. (Ed.) (2022) 'Design of financial risk early warning system in enterprise economic management based on cluster algorithm', *4th International Conference on Decision Science and Management (ICDSM)*, Changsha, pp.511–520, DOI: 10.1007/978-981-19-2768-3_49.

Hwang, B.N., Puntha, P. and Jitanugoon, S. (2025) 'AI-driven circular transformation: unlocking sustainable startup success through co-creation dynamics in circular economy ecosystems', *Sustainable Development*, Vol. 33, pp.245–74, DOI: 10.1002/sd.70001.

Li, J. and Qin, F. (2025) 'Design and development of machine learning based enterprise financial crisis early warning system', *2025 International Conference on Intelligent Computing and Knowledge Extraction-ICICKE*, Bengaluru, India, pp.1–7, DOI: 10.1109/ICICKE65317.2025.11136208.

Li, L., Peng, Y.B., Shao, G.L. and Dai, H.Y. (2025) 'Betting on success: Unveiling the role of innovation and financing capability on funding decisions by human versus AI evaluators', *Economics Letters*, Vol. 256, p.6, DOI: 10.1016/j.econlet.2025.112605.

Li, N. (2021) 'Design of enterprise financial crisis early warning system based on BP neural network', *13th International Conference on Measuring Technology and Mechatronics Automation (ICMTMA)*, Beihai, China, pp.33–36, DOI: 10.1109/ICMTMA52658.2021.00017.

Paeplow, J., Schoormann, T., Möller, F. and Strobel, G. (2025) 'AI startups for good: a taxonomy and archetypes of sustainable business models', *Journal of Cleaner Production*, Vol. 520, p.13, DOI: 10.1016/j.jclepro.2025.146144.

Raneri, S., Lecron, F., Hermans, J. and Fouss, F. (2023) 'Predictions through Lean startup? Harnessing AI-based predictions under uncertainty', *International Journal of Entrepreneurial Behavior & Research*, Vol. 29, No. 4, pp.886–912, DOI: 10.1108/ijebr-07-2021-0566.

Shen, Y.C., Wang, C.H. and Chu, Y.Y. (2025) 'Determinants of artificial intelligence startups: evidence from 150 countries', *Chinese Management Studies*, p.27, DOI: 10.1108/cms-09-2024-0676.

Staub, L., van Giffen, B., Hehn, J. and Sturm, S. (2023) 'Design thinking for artificial intelligence: how design thinking can help organizations to address common AI project challenges', *25th International Conference on Human-Computer Interaction (HCI International)*, Copenhagen, pp.251–267, DOI: 10.1007/978-3-031-48057-7_16.

Takas, N., Kouloumpris, E., Moutsianas, K., Liapis, G., Vlahavas, I. and Kousenidis, D. (2024) 'Startup sustainability forecasting with artificial intelligence', *Applied Sciences-Basel*, Vol. 14, No. 19, p.13, DOI: 10.3390/app14198925.

Triando, S.M., Wang, X. and Prasad, A.S.R. (2024) 'The use of generative ai tools in the inception stage of software startups', *International Conference on Software Business*, Springer Nature Switzerland, Cham, pp.439–453.

Wang, X.N. and Wu, L. (2025) 'Artificial intelligence, lean startup method and product innovations', *Management Science*, p.28, DOI: 10.1287/mnsc.2022.03905.

Wang, Y. and Guan, Q. (2025) 'Studies on innovative energy startups' topological roles and their correlation with success: based on temporal networks', *Frontiers in Physics*, Vol. 13, p.16. DOI: 10.3389/fphy.2025.1610832.

Zhang, J.X. (2024) 'Research on enterprise financial risk early warning system model based on BP neural network', *2024 International Conference on Power, Electrical Engineering, Electronics and Control*, Athens, Greece, pp.599–603, DOI: 10.1109/PEEEC63877.2024.00114.

Zhu, W.D., Zhang, T.J., Wu, Y., Li, S.R. and Li, Z.M. (2022) 'Research on optimization of an enterprise financial risk early warning method based on the DS-RF model', *International Review of Financial Analysis*, Vol. 81, p.9, DOI: 10.1016/j.irfa.2022.102140.