Prescription Recommendation Algorithm Based on Herbal Property-Driven Compatibility Mechanism Semantic Modeling

Xueru Geng[1], Jiantong Zhang[1], Tao Luo[1,*], Jianchen Hou[2], Xiaohua Tao[2,*]

[1]Beijing University of Posts and Telecommunication, School of Information and Communication Engineering, Beijing 100876, China
[2]Beijing University of Chinese Medicine, School of Traditional Chinese Medicine, Beijing 102488, China

Abstract: The in-depth analysis of the semantic information contained in Traditional Chinese Medicine (TCM) prescriptions is of great significance for both clinical applications and the discovery of new formulas. Existing TCM prescription generation algorithms define the interactions between all symptom-herb pairs solely based on co-occurrence, without considering the categorization of herbal properties. To address this issue, this paper proposes a Prescription Recommendation Algorithm based on Herbal Property-Driven Compatibility Mechanism Semantic Modeling (HPDCM). First, the analysis of prescriptions takes into account the herbal property categories, which are defined as entities when constructing the knowledge graph. Second, the algorithm integrates compatibility rules to model the interactions between symptoms and herbs with weighted connections. This is followed by aggregating higher-order heterogeneous path information of nodes through a graph convolutional network model. Finally, an attention mechanism is employed to fuse information from symptom interaction graphs, symptom-herb interaction graphs, and herb interaction graphs, distinguishing the influence of different dimensions of TCM semantic information. Experimental results, compared with existing formula generation algorithms, demonstrate that HPDCM achieves higher accuracy and is more in line with the TCM diagnostic and therapeutic principles of syndrome differentiation and treatment.

1 Introduction

Traditional Chinese Medicine (TCM) is a medical system rooted in ancient Chinese pragmatic materialist thought, focusing on human life processes and disease treatment. TCM follows the diagnostic logic of "Theory - Method - Prescription - Medicine," where doctors first identify the disease cause based on symptoms (differentiating syndromes) [1], then determine the treatment method, and finally select the appropriate prescription. Due to the complexity and subjectivity of syndrome differentiation, TCM relies heavily on the doctor's judgment. With advancements in AI, deep learning-based decision-making models to assist in prescription formulation have become a key area in TCM intelligent healthcare research.

Prescription recommendation algorithms can be categorized into three types: frequency statistics and data mining-based, topic model-based, and deep learning-based approaches. Frequency statistics and data mining methods use co-occurrence to mine therapeutic relationships between symptoms and herbs. Zhang et al. [2] proposed PRCN, identifying core herbs based on their frequency in prescriptions. Ruan et al. [3] applied clustering algorithms to uncover co-occurrence patterns. Topic models transform prescription data into a text corpus to identify herb themes related to symptoms. Yao et al. [4] used topic models to find symptom-herb correlations. However, topic models are limited by their "bag-of-words" nature and struggle with capturing contextual relationships in TCM prescriptions. Both of the first two approaches focus on statistical relationships, failing to capture the complex semantic connections in TCM, making them less effective for complex disease prescriptions.

Neural network-based deep learning recommendation models offer several advantages, such as strong data adaptability, powerful semantic representation capabilities, and robust learning ability. These models are gradually being applied in the field of TCM. For example, Liu et al. [5] combined Text Convolutional Neural Networks (TextCNN) with Long Short-Term Memory (LSTM) to integrate semantic features of symptoms and sequence features of prescription texts, thereby obtaining comprehensive features of symptoms and herbs. Compared to the previous two methods, this approach more comprehensively explores the contextual semantic relationships between symptoms and herbs in prescription texts. However, prescription text data contains a large amount of noise and ambiguous contextual relationships, which reduces the ability of deep learning to extract and associate TCM knowledge, thus affecting the performance of prescription recommendation.

Knowledge graphs represent knowledge in a structured way using triples to describe entities and their

* Corresponding author: tluo@bupt.edu.cn; xhtao1963@126.com

© The Authors, published by EDP Sciences. This is an open access article distributed under the terms of the Creative Commons Attribution License 4.0 (https://creativecommons.org/licenses/by/4.0/).

relationships, minimizing redundancy and noise in text data. In TCM, extracting symptom and herb entities from large-scale prescription data and building a TCM knowledge graph can enhance prescription recommendations when combined with deep learning to uncover latent semantic relationships. For instance, Dong et al. [6] proposed the subnetwork-based symptom term mapping (SSTM) method using meta-paths to represent clinical symptoms. Jin et al. [7] introduced the Syndrome-aware Multi-Graph Convolution Network (SMGCN), which models symptom-herb interactions using Graph Convolutional Networks (GCN). Yang et al. [8] added 27 herbal properties to create a multi-level fusion model, though it overlooked the influence of herbal properties on symptoms. Li et al. [9] developed the KGAPG model, constructing a symptom-herb-properties knowledge graph, but only considered co-occurrence interactions without accounting for compatibility rules. Deep learning-based prescription recommendation algorithms have improved accuracy by extracting TCM entities and building knowledge graphs to uncover relationships between symptoms and herbs. However, these methods mainly focus on herb names and properties, linking symptoms and herbs without incorporating TCM theory and compatibility rules. As a result, recommendations may include contradictions or overlapping effects among herbs, reducing their accuracy and practicality.

Based on related TCM prescription theory [10-12], it is known that the four herbal properties: nature, flavor, tropism, and toxicity are important factors influencing the compatibility of herbs. In response to the lack of in-depth exploration of the principles of TCM prescriptions in existing research, this paper proposes a Prescription Recommendation Algorithm Based on Herbal Property-Driven Compatibility Mechanism Semantic Modeling (HPDCM), which comprehensively considers TCM compatibility rules and clinical treatment principles. The main contributions are as follows:

- We introduce four herb properties—nature, flavor, tropism, and toxicity (39 categories in total)—to construct a symptom-herb-property multi-relational graph for prescription recommendation. Additionally, we build the herb graph based on herb co-occurrence relationships and the symptom graph based on symptom co-occurrence relationships.
- $\mathcal{C}$ (Compatibility Law) represents the coupling of various relationships between herbs and their properties. Symptom-herb interactions are modeled as a weighted fusion of multiple $c_v(c_v \in \mathcal{C})$. Using GCN-MISHP aggregates high-order heterogeneous path information between symptom and herb property nodes in the multi-relational graph to uncover complex compatibility behind symptom-herb interactions. An attention mechanism fuses feature vectors from the symptom graph, multi-relational graph, and herb graph, capturing different dimensions of TCM semantic information.
- Experimental results show that HPDCM achieves precision, recall, and F1 score of 0.2621, 0.1821, and 0.2174, respectively, which are improvements of 10.26%, 8.21%, and 9.08% compared to existing models.

2 System Model

2.1 Problem Description

In the prescription dataset P, let $S=\{s_1,\cdots,s_M\}$ and $H=\{h_1,\cdots,h_N\}$ represent all symptoms and herbs, respectively, where M and N are the total numbers of symptoms and herbs in the dataset. Each prescription data $p(p\in P)$ consists of a symptom set $s_{set}=\{s_1,s_2,\cdots,s_i\}$ and a corresponding herb set $h_{set}=\{h_1,h_2,\cdots,h_j\}$, where $s_i\in S$ and $h_j\in H$. The task of the prescription generation model is to recommend a specific set of herbs based on a given symptom set. The goal is to learn a prediction function

$$\hat{G}(s_{set})=f(s_{set},H\mid\theta) \tag{1}$$

where $\hat{G}(s_{set})$ is an N-dimensional probability vector, and the value in the nth dimension represents the probability of herb h_n contributing to the treatment of the given symptom set s_{set}. The parameter θ represents the trainable parameters of the function $f(\cdot)$.

2.2 HPDCM Overview

Existing algorithms overlook the distinction between herb property categories and treat symptom-herb interactions as simple co-occurrences. This paper improves upon this by incorporating multiple herb properties as different entity types in a multi-relational graph. It defines c_v as the coupling of multi-type relationships in the herb-property triplet, with symptom-herb interaction modeled as a weighted fusion of c_v relationships. We propose the HPDCM algorithm, based on a herb-property-driven compatibility mechanism for semantic modeling. The model architecture is shown in Figure 1.

This paper defines the properties of herbs (flavor, nature, tropism, toxicity) as entities to enhance the directionality and diversity of herb interactions. To address the limitations of existing algorithms in capturing complex symptom-herb compatibility, we construct $\mathcal{G}_{\mathcal{H}}$, an herb external knowledge graph. The paper introduces c_v to couple multiple relationship types in $\mathcal{G}_{\mathcal{H}}$, modeling symptom-herb interactions as a weighted fusion of these c_v. Additionally, we build $\mathcal{G}^1$, a multi-relational knowledge graph for prescription recommendation, incorporating symptoms, herbs, and properties. Following prior models [7-8], we also create $\mathcal{G}^2$ (symptom knowledge graph) and $\mathcal{G}^3$ (herb knowledge graph) to capture symptom co-occurrence

and herb compatibility relationships. The GCN-MISP, GCN-MIHP, and GCN-MISHP are used to learn the node embeddings of $\mathcal{G}^1$, $\mathcal{G}^2$, and $\mathcal{G}^3$, respectively. Attention mechanisms fuse the feature vectors of symptoms and herbs from the three graphs, capturing TCM semantic information across dimensions. The multiple symptoms in each prescription are combined into a symptom set vector, which is then used to calculate the probability of each herb treating the symptom set through an inner product with the herb embeddings.

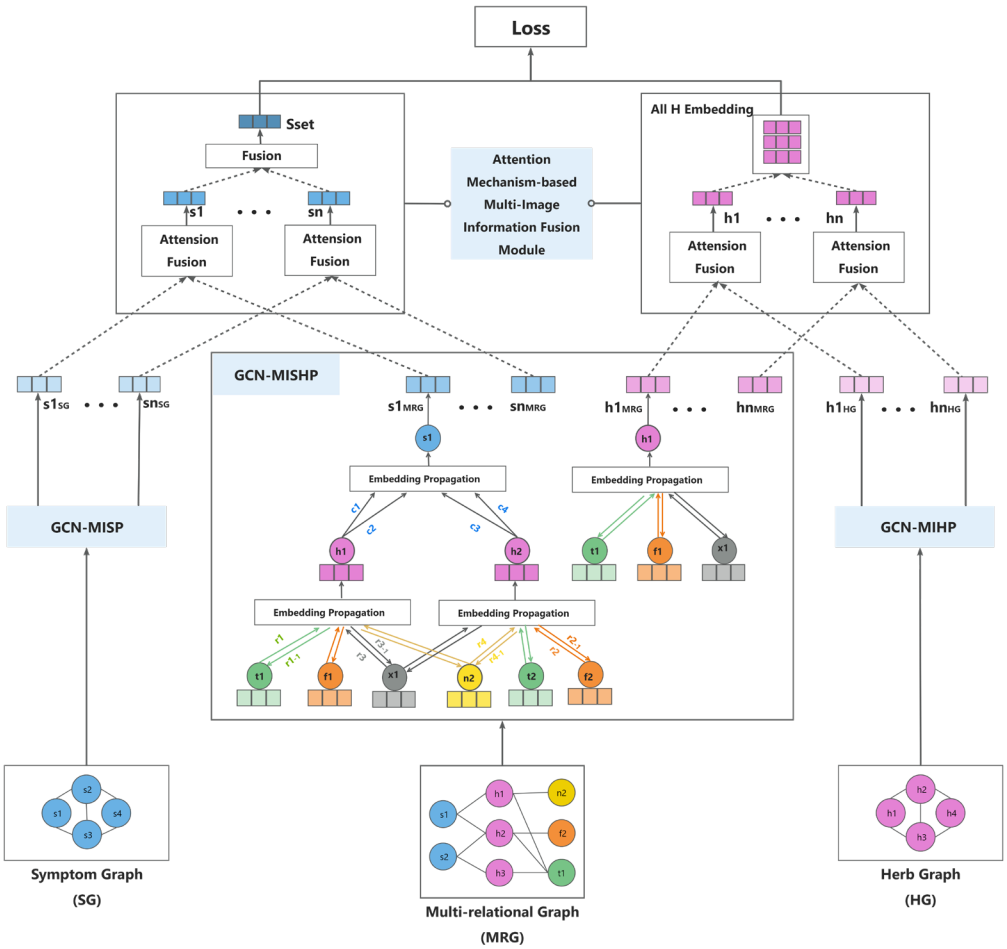

Figure 1. HPDCM Architecture Diagram.

2.3 Multi-Graph Construction

This paper constructs $\mathcal{G}^1$ based on herb compatibility rules, and $\mathcal{G}^2$ and $\mathcal{G}^3$ based on co-occurrence relationships to fully extract TCM knowledge from prescription data. Compatibility relationships are denoted as $\mathcal{C}=\{c_1,\cdots,c_V\}$, where V represents the number of relationships. Herb property nodes are denoted as $\mathcal{Q}=\{q_1,\cdots,q_K\}$, where K represents the number of property nodes.

2.3.1 Construction of $\mathcal{G}^1$

TCM herb properties include nature, flavor, tropism, and toxicity, with compatibility reflecting a complementary relationship between them. For instance, while both Perilla leaf and Scutellaria baicalensis enter the lung tropism, their distinct properties lead to different therapeutic effects—Perilla leaf is warm and pungent, promoting qi and dispersing cold, while Scutellaria baicalensis is cold, clearing heat and draining fire. Existing knowledge graphs often categorize all herbal properties into one node class, which is insufficient. This paper constructs $\mathcal{G}^1$ to better represent herb compatibility and its impact on symptom treatment.

By integrating the symptom-herb interaction set $\mathcal{O}$ with the external herb knowledge graph $\mathcal{G}_{\mathcal{H}}$, we can obtain $\mathcal{G}^1=\{\mathcal{S},\mathcal{R}_1^1,\mathcal{H},\mathcal{R}_1^2,\mathcal{Q}\}$.

Here, $\mathcal{S}(\mathcal{S}\in S)$ represents a set of symptoms, $\mathcal{H}(\mathcal{H}\in H)$ represents a set of herbs, and $\mathcal{O}=\{(s_i,r_{\text{int}},h_j)|s_i\in\mathcal{S},r_{\text{int}}\in\mathcal{R}_1^1,h_j\in\mathcal{H}\}$ is the set of symptom-herb interaction. In this set, r_{int} indicates the co-occurrence of symptom s_i and herb h_j in any prescription from the prescription dataset P. As shown in Table 1, the interaction in (s_i,h_j) is implicit, essentially arising from the complex compatibility of

various herbal properties of the herbs. Therefore, in the construction of the graph, the relationship r_{int} is introduced to explicitly represent the relationship between s_i and h_j. However, the complex modeling process of h_j will be further detailed in the next section through the introduction of the compatibility relationship c_v.

Table 1. Prescription Data Example.

Symptom Set	Herb Set
fever, headache, facial redness, aversion to wind, cough, dry throat, nasal congestion, runny nose, dry mouth, Floating and rapid pulse, thin white tongue coating.	honeysuckle flower, forsythia, gardenia, mint, burdock seed, platycodon, licorice root.

The relationship set between herb nodes and property nodes is defined as $\mathcal{R}_1^2 = \{r_1, \cdots, r_u, \cdots, r_U\}$, where U denotes the number of relationship types, including nature, flavor, tropism, and toxicity. $\mathcal{Q}(\mathcal{Q} \in Q)$ is a set of property entities. Based on this, we can obtain the $\mathcal{G}_{\mathcal{H}} = \{(h_j, r_u, q_k) \,|\, h_j \in \mathcal{H}, r_u \in \mathcal{R}_1^2, q_k \in \mathcal{Q}\}$, which is an external herb knowledge graph. The statistical data of $\mathcal{G}^1$ based on the above content is shown in Table 2.

Table 2. Information Statistics of $\mathcal{G}^1$.

Components	Category	Quantity
$\mathcal{O}$	symptom nodes	360
	herb nodes	753
	(s_i, h_j)	37096
$\mathcal{G}_{\mathcal{H}}$	herb nodes	753
	property nodes	39
	type of r_u	4
	(h_j, r_u, q_k)	7554

2.3.2 Construction of $\mathcal{G}^2$ and $\mathcal{G}^3$

In TCM, primary symptoms represent the main disease manifestations, while secondary symptoms are auxiliary. Comorbidity can occur, where symptoms interact and reflect disease complexity. Herb compatibility in prescriptions is based on the interaction of their properties (e.g., hot, cold, cool), with herbs paired for complementary effects. The statistical data of $\mathcal{G}^2$ and $\mathcal{G}^3$ are shown in Table 3.

To effectively extract the co-occurrence relationships between symptoms, closely aligned with TCM theory, this paper constructs $\mathcal{G}^2 = \{\mathcal{S}, \mathcal{R}^2\}$, where $\mathcal{R}^2 = \{r^2_{s_m, s_{m'}} \,|\, r^2_{s_m, s_{m'}} = \mathbb{1}(\sum_{i=1}^{I} s_{i,m}, s_{i,m'} \geq T_1)\}_{m=1,m'=1}^{M,M}$ describes the relationship between any two symptom nodes in the graph. Specifically, T_1 is the co-occurrence frequency threshold for two symptoms. If the co-occurrence frequency of s_m and $s_{m'}$ in P exceeds T_1, then $r^2_{s_m, s_{m'}} = 1$, otherwise, $r^2_{s_m, s_{m'}} = 0$.

To effectively extract the co-occurrence relationships between herbs and explore prescription patterns, this paper constructs $\mathcal{G}^3 = \{\mathcal{H}, \mathcal{R}^3\}$, where $\mathcal{R}^3 = \{r^3_{h_n, h_{n'}} \,|\, r^3_{h_n, h_{n'}} = \mathbb{1}(\sum_{i=1}^{I} h_{i,n}, h_{i,n'} \geq T_2)\}_{n=1,n'=1}^{N,N}$ dscribes the relationship between any two herb nodes in the graph. Specifically, T_2 is the co-occurrence frequency threshold for two herbs. If the co-occurrence frequency of h_n and $h_{n'}$ in P exceeds T_2, then $r^3_{h_n, h_{n'}} = 1$, otherwise, $r^3_{h_n, h_{n'}} = 0$.

Table 3. Information Statistics of $\mathcal{G}^2$ and $\mathcal{G}^3$.

Graph	Number of Nodes	Number of Edges	Average Degree of Nodes
$\mathcal{G}^2$	360	1916	5.32
$\mathcal{G}^3$	753	6819	9.05

2.4 Embedding Learning

2.4.1 Information Aggregation of GCN-MISHP

$\mathcal{G}^1$ includes herb-property and symptom-herb interactions. To capture the distinct Chinese medicine semantics, this paper uses GCN-MISHP to learn the embedding representations of symptom, herb, and property nodes.

In herb-property pairs, herbs may have multiple properties with varying contributions, and different herbs with the same property may show different contributions. Thus, the same herb (or property) can appear in multiple triples (h_j, r_u, q_k), carrying different semantic meanings based on context. To model these relationships, four types of r_u relationships are used to distinguish herb properties (Section 2.3). An inverse relationship r_u^{-1} is introduced during node aggregation, revealing the similarities and differences between herbs. The first-order connectivity of herb h_j is represented as $\mathcal{N}_{h_j} = \{(r_u, q_k) \,|\, (h_j, r_u, q_k) \in \mathcal{G}_{\mathcal{H}}\}$, and the first-order connectivity representation of h_j is computed by integrating the relationship-aware information from the first-order connected entities as follows:

$$\boldsymbol{e}_{h_j}^{(1)} = \frac{1}{\left|\mathcal{N}_{h_j}\right|} \sum_{(r_u, q_k) \in \mathcal{N}_{h_j}} \boldsymbol{e}_{r_u} \odot \boldsymbol{e}_{q_k}^{(0)} \tag{2}$$

where $\boldsymbol{e}_{q_k}^{(0)}$ is the initial embedding of the property node q_k, and $\boldsymbol{e}_{r_u}$ is the initial embedding of the relationship r_u, with $\odot$ representing the Hadamard product. Similarly, by aggregating the herb entities connected to a property entity, we can uncover the influence of the

property on different herbs. The first-order connectivity of property q_k is represented as:

$$\boldsymbol{e}_{q_k}^{(1)} = \frac{1}{\left|\mathcal{N}_{q_k}\right|} \sum_{(r_u^{-1}, h_j) \in \mathcal{N}_{q_k}} \boldsymbol{e}_{r_u^{-1}} \odot \boldsymbol{e}_{h_j}^{(0)} \tag{3}$$

where $\boldsymbol{e}_{h_j}^{(0)}$ is the initial embedding of the herb node h_j, and represents the first-order connectivity of h_j, and $\mathcal{N}_{q_k}$ represents the first-order connectivity of h_j.

Next, in symptom-herb interaction pairs, each interaction r_{int} between a symptom and an herb contains different semantics. To finely model the unique TCM semantics of each $(s_i, r_{\text{int}}, h_j)$ interaction, we define a compatibility rule c_v as a weighted fusion of the feature vectors of relationships r_u between all herb and property nodes. This represents the interaction between symptom-herb pairs driven by multiple property compatibilities. The embedding representation of c_v is computed as:

$$\boldsymbol{e}_{c_v} = \sum_{r_u \in \mathcal{R}_1^2} \frac{\exp(\omega_{r_u c_v})}{\sum_{r_u' \in \mathcal{R}_1^2, r_u' \neq r_u} \exp(\omega_{r_u' c_v})} \boldsymbol{e}_{r_u} \tag{4}$$

where r_u' is any relationship type different from r_u in $\mathcal{R}_1^2$, and $\omega_{r_u c_v}$ is a trainable weight that quantifies the importance of each relationship r_u to the c_v. Based on the diversity of compatibility relations, the types of c_v can be manually assigned, and the optimal value is sought during model training. To ensure that each c_v carries unique compatibility rule information, we introduce a distance-related function [13]:

$$\mathcal{L}_{IND} = \sum_{\mathrm{c}_v, \mathrm{c}_v' \in C, \mathrm{c}_v \neq \mathrm{c}_v'} \frac{dCov(\boldsymbol{e}_{c_v}, \boldsymbol{e}_{c_v'})}{\sqrt{dVar(\boldsymbol{e}_{c_v}) \bullet dVar(\boldsymbol{e}_{c_v'})}} \tag{5}$$

where $dCov(\cdot)$ is the distance covariance between two vectors, and $dVar(\cdot)$ is the distance variance. By minimizing $\mathcal{L}_{IND}$, we can maximize the dissimilarity between different c_v. To capture the potential complex semantics behind each symptom-herb interaction, we fine-grain the modeling of r_{int} using c_v. For a particular symptom, different c_v have varying degrees of influence on the selection of herbs. Therefore, we introduce an attention score

$$\alpha(s_i, c_v) = \frac{\exp(\boldsymbol{e}_{c_v}^{\mathrm{T}} \boldsymbol{e}_{s_i}^{(0)})}{\sum_{\mathrm{c}_v, \mathrm{c}_v' \in C, \mathrm{c}_v \neq \mathrm{c}_v'} \exp(\boldsymbol{e}_{c_v'}^{\mathrm{T}} \boldsymbol{e}_{s_i}^{(0)})} \tag{6}$$

to model the embedding representation of r_{int} as:

$$\boldsymbol{e}_{r_{\text{int}}} = \alpha(s_i, c_v) \boldsymbol{e}_{c_v} \tag{7}$$

Thus, the first-order connectivity representation of symptom s_i is given as:

$$\boldsymbol{e}_{s_i}^{(1)} = \frac{1}{\left|\mathcal{N}_{s_i}\right|} \sum_{(r_{\text{int}}, h_j) \in \mathcal{N}_{s_i}} \boldsymbol{e}_{r_{\text{int}}} \odot \boldsymbol{e}_{h_j}^{(0)} \tag{8}$$

where $\mathcal{N}_{s_i}$ represents the first-order connectivity of symptom s_i. After modeling the first-order connectivity for both h_j and s_i nodes, additional aggregation layers are added to collect information from higher-order neighbors. Let path $l = s_1 \xrightarrow{\gamma_1} \cdots \xrightarrow{\gamma_n} l_n$ be an n-hop path rooted at symptom node s_i, where the path consists of a sequence of relationships $(\gamma_1, \cdots, \gamma_n)$. The n-order connectivity representation of s_i is then:

$$\boldsymbol{e}_{s_i}^{(l)} = \sum_{l \in \mathcal{N}_{s_i}^l} \frac{\boldsymbol{e}_{\gamma_1}}{\left|\mathcal{N}_{l_1}\right|} \odot \frac{\boldsymbol{e}_{\gamma_2}}{\left|\mathcal{N}_{l_2}\right|} \odot \cdots \odot \frac{\boldsymbol{e}_{\gamma_n}}{\left|\mathcal{N}_{l_n}\right|} \odot \boldsymbol{e}_{l_n}^{(0)} \tag{9}$$

where $\mathcal{N}_{s_i}^l$ is the set of all n-hop paths starting from s_i, and γ_n is any relationship type in the set of all relationships between nodes. This representation learns the interaction between multiple relationships in the path and retains the overall semantics of the path. Similarly, the n-order connectivity representation of h_j is:

$$\boldsymbol{e}_{h_j}^{(l)} = \sum_{l \in \mathcal{N}_{h_j}^l} \frac{\boldsymbol{e}_{\gamma_1}}{\left|\mathcal{N}_{l_1}\right|} \odot \frac{\boldsymbol{e}_{\gamma_2}}{\left|\mathcal{N}_{l_2}\right|} \odot \cdots \odot \frac{\boldsymbol{e}_{\gamma_n}}{\left|\mathcal{N}_{l_n}\right|} \odot \boldsymbol{e}_{l_n}^{(0)} \tag{10}$$

where $\mathcal{N}_{h_j}^l$ is the set of all n-hop paths starting from h_j. After the n-order information aggregation, we obtain the final embedding representation of symptom s_i and herb h_j in $\mathcal{G}^1$, represented as:

$$\boldsymbol{e}_{s_i}^{\mathcal{G}^1} = \boldsymbol{e}_{s_i}^{(1)} + \cdots + \boldsymbol{e}_{s_i}^{(l)} \tag{11}$$

$$\boldsymbol{e}_{h_j}^{\mathcal{G}^1} = \boldsymbol{e}_{h_j}^{(1)} + \cdots + \boldsymbol{e}_{h_j}^{(l)} \tag{12}$$

2.4.2 Information Aggregation of GCN-MISP and GCN-MIHP

For any symptom (or herb) node in $\mathcal{G}^2$ (or $\mathcal{G}^3$), the information of all first-order neighbors is aggregated, and then passed through a linear transformation layer and an activation layer to obtain the final embedding of the symptom (or herb) node as follows:

$$\boldsymbol{e}_{s_i}^{\mathcal{G}^2} = \tanh\Big(\sum_{s_i' \in \mathcal{N}_S^{s_i - s_i}} \boldsymbol{e}_{s_i'}^{(0)} \omega_1\Big) \tag{13}$$

$$\boldsymbol{e}_{h_i}^{\mathcal{G}^3} = \tanh\Big(\sum_{h_j' \in \mathcal{N}_H^{h_j - h_j'}} \boldsymbol{e}_{h_j'}^{(0)} \omega_1\Big) \tag{14}$$

Here, $\mathcal{N}_S^{s_i - s_i'}$ represents the set of symptoms in $\mathcal{G}^2$ where $r_{s_i, s_i'}^2 = 1$, and $\mathcal{N}_H^{h_j - h_j'}$ represents the set of herbs in $\mathcal{G}^3$ where $r_{h_j, h_j'}^3 = 1$. $\boldsymbol{e}_{s_i'}^{(0)}$ and $\boldsymbol{e}_{h_j'}^{(0)}$ are the initial embeddings of the symptom node in $\mathcal{G}^2$ and the herb node in $\mathcal{G}^3$, respectively. ω_1 is the training weight parameter for the first-order neighbors of the current node, which represents the contribution of the neighbors to the feature update of the current node.

2.5 Multiple graphs Information Fusion

To effectively capture these dimensions, the symptom

and herb information from each graph are treated separately. An attention mechanism is used to fuse the node embeddings from each graph, resulting in comprehensive symptom and herb embeddings:

$$\boldsymbol{e}_{s_i}^{*} = \lambda_1 \boldsymbol{e}_{s_i}^{\mathcal{G}^1} + (1-\lambda_1)\boldsymbol{e}_{s_i}^{\mathcal{G}^2} \tag{15}$$

$$\boldsymbol{e}_{h_j}^{*} = \lambda_2 \boldsymbol{e}_{h_j}^{\mathcal{G}^1} + (1-\lambda_2)\boldsymbol{e}_{h_j}^{\mathcal{G}^3} \tag{16}$$

Here, λ_1 and λ_2 are trainable weight parameters.

2.6 Prescription Recommendation Training Process

Define $\boldsymbol{P}$ as the one-hot encoding matrix of s_{set} in the prescription. $\boldsymbol{Z}_s = \{e_{s_1}, \cdots, e_{s_M}\}$ represents the matrix of all symptom embeddings, where $avgpool(\cdot)$ is the average pooling layer and $ReLU(\cdot)$ is the activation function. The embedding of s_{set} is obtained through the following equation:

$$\boldsymbol{e}_{s_{set}} = ReLU(MLP(avgpool(\boldsymbol{P} \bullet \boldsymbol{Z}_s))) \tag{17}$$

Define $\boldsymbol{A}_H = \{e_{h_1}, \cdots, e_{h_N}\}$ as the matrix of all herb embedding vectors. The matrix multiplication of $\boldsymbol{e}_{s_{set}}$ and $\boldsymbol{A}_H$ is computed, and the result is passed through $Sigmoid(\cdot)$ to convert it into a probability value:

$$\hat{G}(\mathrm{s}_{set}) = Sigmoid(\boldsymbol{e}_{s_{set}} \bullet \boldsymbol{A}_H) \tag{18}$$

Finally, herbs within TOP@K based on the probability values are recommended as the prescription.

The recommendation process loss $\mathcal{L}_{MPDCM}$ consists of three components: the multi-cross entropy loss $BCELoss$ [14], $\mathcal{L}_{IND}$, and the regularization term. $BCELoss$ is used to reconstruct the prescription data, for a given s_{set}, the herbs provided by the existing prescription should receive a higher predicted probability than those not provided:

$$BCELoss = -\frac{1}{\mathcal{N}_h} \sum_{h_j \in \mathcal{N}_h} \log \hat{y}(s_{set}, h_j) \tag{19}$$

where $\mathcal{N}_h$ is the set of herbs that treat s_{set} in P. $\mathcal{L}_{MPDCM}$ is defined as:

$$\mathcal{L}_{MPDCM} = BCELoss + \mathcal{L}_{IND} + \|\Theta\|_2^2 \tag{20}$$

Here, $\Theta = \{\boldsymbol{e}_{s_i}^{(0)}, \boldsymbol{e}_{h_j}^{(0)}, \boldsymbol{e}_{q_k}^{(0)}, \boldsymbol{e}_{r_u}, \boldsymbol{e}_{c_v}, \omega_{r_u,c_v}, \omega_1, \lambda_1, \lambda_2\}$ is the set of model parameters, which is optimized using the Adam algorithm [15].

3 Experimental Design and Results Analysis

3.1 Dataset

The TCM benchmark dataset from Jin et al. [9] is used, with added herbal properties. It contains 36,260 prescriptions, 360 symptoms, 753 herbs, and 39 properties. Based on this, we construct the external herb knowledge graph $\mathcal{G}_{\mathcal{H}}$, which includes entity and relationship types as shown in Table 4. A local visualization of $\mathcal{G}_{\mathcal{H}}$ is shown in Figure 2.

Table 4. Information Statistics of $\mathcal{G}_{\mathcal{H}}$.

Components		Category
entity	herb	753
	nature	9: slightly cool, cool, slightly cold, cold, neutral, sightly warm, warm, hot, very hot.
	flavor	14: sour, bitter, sweet, spicy, salty, bland, astringent, slightly sour, slightly bitter, slightly sweet, slightly pungent, slightly salty, slightly astringent, slightly sweet.
	tropism	12: lung, large intestine, stomach, spleen, heart, small intestine, bladder, kidney, pericardium, triple Burner, liver, gallbladder.
	toxicity	4: highly toxic, toxic, slightly toxic, non-toxic
relationship types		4: nature, flavor, tropism, toxicity.

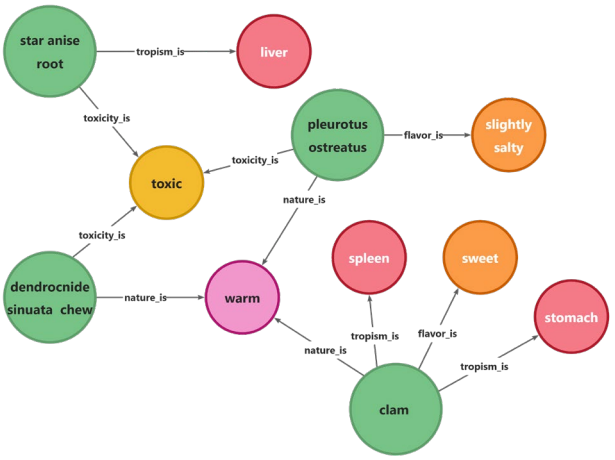

Figure 2. $\mathcal{G}_{\mathcal{H}}$ Local Visualization.

3.2 Evaluation Metrics

We use Precision@K, Recall@K, and F1-Score@K as the evaluation metrics for the model, where K refers to the TOP@K items in the recommended herb list.

3.3 Baseline Models

- LightGCN [16]: Simplifies GCN by focusing on neighbor aggregation, learning embeddings through linear propagation, and using a weighted sum of all layers for final embeddings.
- SMGCN [7]: Generates symptom and herb embeddings with GCN, integrates symptoms into syndromes, and recommends herbs by interacting herb embeddings with syndrome representations.
- KGAPG [9]: Builds a knowledge graph to capture semantic relationships between symptoms and herbs, using attention-based aggregation to simulate the TCM diagnosis process.
- KDHR [8]: Incorporates 27 herb properties as auxiliary information, using multi-layer fusion

graph convolution to generate low-noise symptom node representations.

KDHR_39: It extends KDHR by introducing 39 herb properties using one-hot encoding, while keeping other aspects unchanged.

3.4 Experimental Setup

The threshold for constructing $\mathcal{G}^2$ and $\mathcal{G}^3$ are set to 5 and 40, respectively, based on the baseline model. The dataset is split into training, validation, and test sets (6:2:2 ratio). The learning rate for HPDCM, KDHR, and SMGCN is 0.0003, with a regularization coefficient of 0.007. For other models, learning rates are 0.001 and 0.0001, decaying every 7 epochs. All models train for 200 epochs with a batch size of 512. Embedding and hidden layer dimensions are 64, and the final output dimension is 256. For $\mathcal{G}^2$ and $\mathcal{G}^3$, GCN convolution layers are set to 1. Early stopping is applied if validation loss doesn't improve for 7 epochs.

3.5 Result Analysis

3.5.1 Effectiveness of HPDCM

The gains in Table 5 are calculated by comparing the optimal results of HPDCM (bold and underlined) with those of the baseline models.

Table 5. Performance Comparison.

Model	Precision			Recall			F1-Score		
	P@5	P@10	P@20	R@5	R@10	R@20	F1@5	F1@10	F1@20
LightGCN	0.2270	0.1710	0.1279	0.1557	0.2344	0.3504	0.1847	0.1977	0.1874
SMGCN	0.2285	0.1676	0.1144	0.1559	0.2259	0.3048	0.1853	0.1925	0.1664
KGAPG	0.2295	0.1735	0.1270	0.1598	0.2378	0.3485	0.1884	0.2006	0.1862
KDHR	0.2309	0.1814	0.1340	0.1662	0.2587	0.3779	0.1933	0.2133	0.1979
KDHR_39	**0.2377**	**0.1853**	**0.1367**	**0.1716**	**0.2638**	**0.3858**	**0.1993**	**0.2177**	**0.2019**
HPDCM	0.2621	0.2062	0.1507	0.1857	0.2912	0.4244	0.2174	0.2415	0.2224
Gains	10.26%	11.27%	10.24%	8.21%	10.38%	10.00%	9.08%	10.93%	10.15%

HPDCM outperforms all models in three evaluation metrics, aligning best with the TCM diagnostic and treatment process. SMGCN performs the worst due to the lack of herbal property knowledge. LightGCN improves on SMGCN by reducing redundancy but fails to extract deep TCM insights. KGAPG and KDHR outperform SMGCN and LightGCN by incorporating herbal properties but treat all properties equally, missing deeper symptom-herb interactions. Training curves for precision, recall, and F1 score in Figure 3 show that after 100 epochs, performance stabilizes, indicating HPDCM effectively learns key features from prescription data for accurate recommendations.

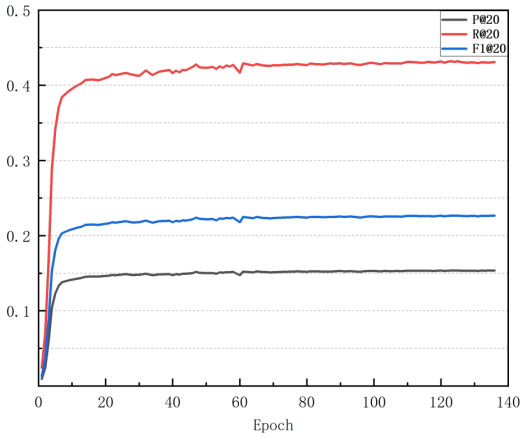

Figure 3. The training curves of evaluation metrics.

3.5.2 The Impact of Single Symptom and Multiple Symptoms Datasets

Herb recommendation in TCM syndrome differentiation relies on multi-symptom data, as single-symptom data are scarce in practice. Evaluating on both single- and multi-symptom datasets (26,360 records), results in Figure 4 show HPDCM performs better on multi-symptom data, reflecting the need for multiple symptoms in syndrome differentiation. This demonstrates HPDCM's effectiveness in learning syndrome differentiation and mitigating inaccuracies from missing symptoms.

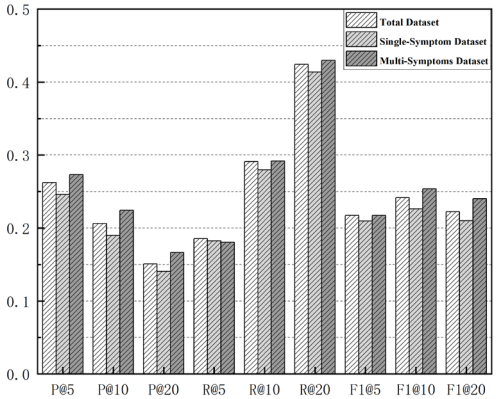

Figure 4. Performance Comparison Across Different Datasets.

3.5.3 Ablation Experiment

An ablation experiment was conducted to assess the attention mechanism's role in multi-graph information NoAtt refers to the original model without the attention module. Results in Table 6 show that the attention mechanism effectively captures the unique TCM semantics in each graph, highlighting the need for non-linear aggregation to improve recommendation performance.

Table 6. Ablation Experiment Metric Comparison.

Model	Precision@20	Recall@20	F1-Score@20
HPDCM	0.1507	0.4244	0.2224
NoAtt	0.1491	0.4210	0.2202

3.5.4 The Impact of the Number of c_v

As explained in Section 2.4.1, HPDCM models symptom-herb interactions using weighted combinations of manually set TCM compatibility rules. Varying the number of rules ({1, 2, 4, 8, 10}) highlights HPDCM's advantages, as shown in Figure 5. Performance is poor when $c_v = 1$ but improves as c_v increases, indicating that symptom-herb interactions depend on rule combinations. However, when c_v reaches 10, performance declines due to overfitting and conflicts between rules, as the introduction of $\mathcal{L}_{IND}$ maintains rule independence. HPDCM incorporates the four properties of herbs, simulating TCM practitioners' decision-making for more personalized and accurate recommendations.

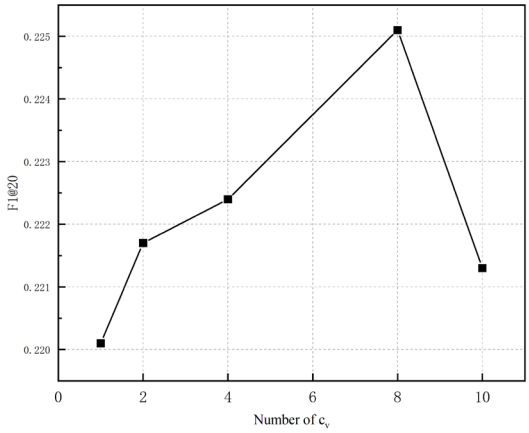

Figure 5. Trend of the Effect of Different c_v.

3.5.5 The Impact of Path Length L

As discussed in Section 2.4.1, higher-order relational paths introduce noise during path aggregation. We examine the effect of different path lengths (L = 1, 2, 3) on model performance, with results shown in Figure 6. L-2 outperforms L-1 by effectively integrating information from higher-order neighbors, capturing more pharmacological features and better simulating TCM diagnosis. In contrast, L-3 performs the worst, as it only connects pharmacological nodes as first-order neighbors, revealing the TCM herb dataset's limitation in capturing intrinsic pharmacological relationships, which introduces noise and degrades performance. Longer relational paths reduce accuracy and affect recommendations.

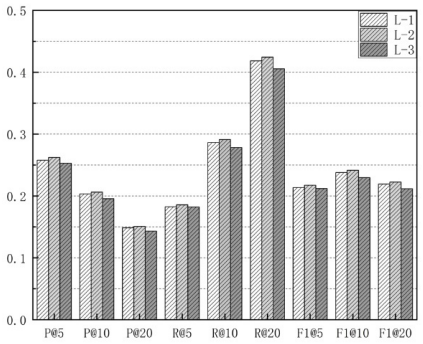

Figure 6. Performance Comparison Across Different L.

4 Summary and Future Work

This paper presents the HPDCM algorithm for prescription recommendation, modeling symptom-herb interactions to reveal herb compatibility patterns and enhance feature representations. It aligns with TCM diagnosis and treatment, showing advantages on TCM datasets. Future work will focus on integrating syndrome information, as the current dataset includes only primary symptoms, which may affect syndrome inference accuracy. Ensuring alignment with TCM's syndrome differentiation theory is a key direction for further research.

References

1. Q. Zhang, X. Zhou, China J. Tradit. Chin. Med. Pharm., **36**, 5 (2021).
2. Y.D. Zhang, K.F. Hu, T. Yang, J.D. Xie, G.L. Shen, Lishizhen Med. Mater. Med. Res., **30**, 5 (2019).
3. C. Ruan, Y. Wang, Y. Zhang, Y. Yang, *Exploring regularity in traditional Chinese medicine clinical data using heterogeneous weighted networks embedding* (IEEE Press, Piscataway, 2019).
4. L. Yao, Y. Zhang, B. Wei, W. Zhang, Z. Jin, IEEE Trans. Knowl. Data Eng., **30**, 6(2018).
5. D. Liu, Y. Zhang, L. Wang, *A TCM therapeutic methods based on text convolutional neural network* (IEEE Press, Piscataway, 2021).
6. X. Dong, Y. Zheng, Z. Shu, K. Chang, D. Yan, *TCMPR: TCM Prescription recommendation based on subnetwork term mapping and deep learning. BioMed Research International* (IEEE Press, Houston, 2021).
7. Y. Jin, W. Zhang, X. He, X. Wang, X. Wang, *Syndrome-aware herb recommendation with multi-graph convolution network* (IEEE, 2020).
8. Y. Yang, Y. Rao, M. Yu, Y. Kang, Neural Network, **146** (2022).
9. S. Li, W. Wang, J. He, KGAPG: *Knowledge-Aware Neural Group Representation Learning for Attentive Prescription Generation of Traditional Chinese Medicine* (IEEE Press, Houston, 2021).
10. F. Zhang, Y. Lin, L. Le, China J. Exp. Tradit. Med. Formulae, **26**, 13(2020).

11. S. Hu, W. Hou, P. Ha, J. Li, S. Chu, J. Tradit. Chin. Med., **62**, 6(2021).
12. F. Wang, J. Tradit. Chin. Med. Manag., **22**, 3(2014).
13. X. Wang, H.Y. Jin, A. Zhang, X.N. He, T. Xu, *Disentangled Graph Collaborative Filtering* (ACM, New York, 2020).
14. U.M. Braga-Neto, *Fundamentals of pattern recognition and machine learning* (Springer, Heidelberg, 2020).
15. D.P. Kingma, J.L. BA, *Adam: A method for stochastic optimization* (ICLR, Washington, 2015).
16. X. He, K. Deng, X. Wang, Y. Li, Y. Zhang, *LightGCN: Simplifying and Powering Graph Convolution Network for Recommendation* (ACM, New York, 2020).