systems

MDPI

Article

Recommendation Method Based on Heterogeneous Information Network and Multiple Trust Relationship

Chun Yan [1,2,*] and Lu Liu [1]

[1] College of Economics and Management, Shandong University of Science and Technology, Qingdao 266590, China; luliu@sdust.edu.cn
[2] College of Mathematics and Systems Science, Shandong University of Science and Technology, Qingdao 266590, China
* Correspondence: yanchun@sdust.edu.cn

Abstract: A recommendation method based on heterogeneous information networks and multiple trust relationships is proposed. Firstly, the node sequence in the heterogeneous information network is obtained through the random walk of the meta-path, and the representation vector of each node in different paths is generated. The user similarity based on the meta-path is obtained by calculating the spatial distance between the user node vectors. Then, according to the different relationships among users, different trust-relationship calculation methods are proposed, and the user similarity based on the user's multiple trust relationship is obtained by fusing multiple trust relationships. Finally, the candidate list of Microblog text is obtained by fusing the two user similarities to achieve a personalized recommendation of Microblog text. The experimental results show that the method proposed in this study is superior to other comparison algorithms in precision, recall, F1 value and NDCG value, which shows that this method is feasible when recommending Microblog text.

Keywords: meta-path; random walk; trust relationship; recommendation

Citation: Yan, C.; Liu, L. Recommendation Method Based on Heterogeneous Information Network and Multiple Trust Relationship. *Systems* **2023**, *11*, 169. https://doi.org/10.3390/systems11040169

Academic Editor: Colette Rolland

Received: 1 February 2023
Revised: 16 March 2023
Accepted: 18 March 2023
Published: 24 March 2023

1. Introduction

In the era of Web 2.0, information resources show an explosive growth trend, and the traditional search methods that rely on the manual input of query conditions can no longer meet people's increasingly diverse needs. As one of the typical representatives derived from the era of Web 2.0, the development and application of Microblog has had a huge impact on Internet information dissemination and social production and lifestyle, and has gradually become an interactive platform for social information sharing and emotional expression.

Microblog originates from blog, but it is different from a traditional blog, which is shown in the following aspects: (1) the number of characters that Microblog users publish information with must be controlled within 140, and cannot be as long as a blog; (2) the publishing methods of Microblog are more convenient and diversified. That is, users can publish Microblog information through computers, mobile phone clients and in other ways; (3) Microblog can provide real-time communication between users anytime and anywhere, which is more real-time, convenient and fast than a traditional blog. In addition, on the Microblog platform, users have formed a complex interpersonal network through "follow" and "followed", and the information released by users has also spread rapidly on the network in a manner similar to a "virus", through forwarding, pushing and other forms. This makes Microblog not only expand the user's interpersonal circle to achieve social interaction, but also become an important medium to obtain real-time information and multiple comments. Therefore, in recent years, Microblogging has attracted more and more people's attention and love, and many blog users have abandoned blogs to join the ranks of Microblogging. Microblog user data can be roughly divided into three parts, which are Microblog user's own data, Microblog user's behavior data and Microblog user's text data. Among these, Microblog user's own data refers to user ID, name, gender, profile, region

and other basic information; Microblog-user behavior data refers to a series of behaviors of users to publish, comment, forward and like Microblog content; Microblog-user text data refers to the text content that users use to generate such behaviors as publishing, commenting, forwarding and liking.

Compared with traditional media, social network platforms such as Microblog can provide enterprises with more convenient and effective means of communication when disseminating marketing information, and improve the scope, speed and duration of communication [1]. At the same time, various social interaction behaviors of Microblog users [2–4] can reflect the potential preferences of customers. Through the analysis of users' social network behaviors [5–7], it creates conditions for enterprises to deeply explore the potential needs of customers. With the rapid growth of Microblog users and Microblog information, a small amount of useful key information is submerged in the sea of massive information, and the phenomenon of information overload is becoming more prominent. It becomes more difficult for users to find their own interesting content among the massive Microblog information. How to recommend high-quality information content for users among the massive amount of Microblog information, reduce the cost of users in obtaining useful information, solve the problem of "information lost", meet users' personalized information needs, and improve the efficiency of information consumption and utilization has become the primary problem faced by the operation and management of the Microblog platform.

2. Literature Review

As an important technology to solve the problem of information overload, personalized recommendation technology is increasingly favored and supported by people. According to the input of the recommendation model, the recommendation system can be divided into three categories: a content-based recommendation system (using only the descriptive information of users and items), a collaborative filtering recommendation system (using only the interactive information between users and items) and a hybrid-recommendation system (using both the interactive information between users and items and the descriptive information related to users and items).

2.1. Content-Based Recommendation System

The core idea of a content-based recommendation system is to make full use of the content information of items and implement the recommendation system based on information-retrieval [8] and information-filtering [9] technologies. For example, such methods usually use the word frequency–inverse document frequency (TF-IDF) [10], applied in the field of information retrieval to represent text information, vectorize it, and then describe user preferences and item attributes, in turn, and recommend users by calculating the similarity between the description of products to be recommended and user preferences [11–13], Li et al. (2010) used a content based recommendation method to construct a user interest model by extracting text features, and calculated the similarity between candidate news and user interests to obtain recommendation results [14]. Jiang et al. (2014) used fuzzy representation and a diversity selection algorithm to improve the content recommendation methods. The results show that this method solved the problem of recommendation diversity to a certain extent, and the recommendation quality was also improved [15]. When recommending real-time traffic information, Lei et al. (2017) used the content-based method to obtain the support of interest points, used the real-time traffic-network status to obtain the real-time traffic-network support, and integrated the two support levels to achieve the optimization of the recommendation results [16]. It can be seen from the method implementation process that the products recommended by these systems are "excessively similar". At the same time, with the development of the Internet, many scenarios need recommendation systems to filter massive amounts of information, but not every scenario contains a large amount of text information. Therefore, another recommendation strategy came into being—a collaborative filtering system.

2.2. Recommendation System Based on Collaborative Filtering

Compared with the content-based filtering recommendation, the collaborative-filtering recommendation system does not need text description information. It can recommend users based only on the user's interaction history of items. According to different strategies of recommendation methods, recommendation systems based on collaborative filtering can be further divided into the memory-based collaborative-filtering algorithm (memory-based CF) and the model-based collaborative-filtering algorithm (model-based CF).

The memory-based collaborative algorithm [17–21] belongs to a kind of heuristic algorithm. Its core idea is very intuitive, that is, to take advantage of the similarity between user preferences or the attributes of items. Based on this discovery, the algorithm regards the user-history scoring record as a whole as a scoring matrix, and uses the similarity between users or items in the scoring matrix and the existing scores in the scoring matrix to predict the missing scores in the scoring matrix. The advantage of a memory-based collaborative-filtering algorithm is that the model is simple to implement and does not require a lot of parameter training. However, in practical application scenarios, the huge data scale of users and items makes the intuitively constructed scoring matrix often have the characteristics of being large scale and extremely sparse [22]. Therefore, the recommendation effect of memory-based collaborative filtering based only on similarity is often greatly affected in the face of sparse data sets. Machine-learning or data-mining models are used to design collaborative-filtering algorithms [23–30], which are collectively referred to as model-based collaborative-filtering algorithms. By means of machine learning, reducing the dimension of the matrix is a common strategy of model-based collaborative filtering to alleviate the sparsity problem; for example, using the singular value decomposition (SVD) of matrix dimensionality-reduction technology to remove unimportant or unrepresentative users and items, thus reducing the dimension of the user-item scoring matrix. Similarly, Goldberg et al. (1999) proposed the Eigentaste method, which uses principal component analysis to filter users and items, thus reducing the matrix dimension of users' items [31]. Although these methods can effectively reduce the dimension of the user-item scoring matrix, if certain specific users and items are deleted from the records by the system, the relevant important information will also be deleted, thus affecting the effect of the recommendation system.

Collaborative filtering is the most widely used recommendation algorithm. Its main idea is to discover the relevance among users based on their preferences for goods, and to recommend based on the relevance; that is, users with high similarity often have similar preferences. However, for the traditional collaborative-filtering algorithm, the sparsity of data and the cold start problem during recommendation are important issues that cannot be ignored. In the existing relevant literature, many scholars have proposed improvement methods for this problem. Gao et al. (2017) combined collaborative filtering with matrix decomposition methods to obtain user potential characteristics from different dimensions, which effectively alleviated the cold start problem [32]. In order to solve the problem of data sparsity, Gu et al. (2018) integrated community structure and personal attributes into traditional collaborative-filtering algorithms, and achieved good recommendation results [33]. Wei Ling et al. (2020) combined user profile with collaborative filtering to build a hybrid-recommendation algorithm that improved the recommendation accuracy to a certain extent [34]. Zhang Jie et al. (2021) added the penalty factor between users into the Pearson similarity calculation, and fused the results with the scoring information entropy, effectively solving the data sparsity problem [35]. Many researchers have specially designed recommendation systems that are suitable for certain structured auxiliary information, such as social recommendation [36], location-based social recommendation [37], and knowledge-map-based recommendation [38]. These methods have been proven to significantly improve the performance of recommendation systems, but they are often related to specific types of auxiliary information and are not universal. How to design a modeling method that can integrate various types of auxiliary information in the recommendation system, so as to use the auxiliary information more flexibly and effectively to alleviate the

data sparsity and cold start problem in the recommendation system, has become a major research challenge for the recommendation system.

2.3. Hybrid-Recommendation System

In addition to the above two ways of alleviating the sparsity problem by reducing the dimension of the matrix, integrating the content information into the model-based collaborative-filtering method is another common way to alleviate the sparsity. Such methods are called hybrid-recommendation models. For example, Zhang et al. (2015) proposed a hybrid-recommendation algorithm based on network and tag, which only considers the structure between users and items in the network-based reasoning algorithm, and extracts rich information of personalized preferences and item content from collaborative tags. The experimental results show that this method can significantly improve the accuracy, diversity and personalization of recommendations [39]. Yang et al. (2018) proposed an improved hybrid recommendation algorithm based on stack noise reduction and self coding, significantly improving the accuracy of recommendation results [40]. Yuan et al. (2019) proposed a hybrid-recommendation algorithm based on attention mechanism, which uses multiple self-attention mechanisms to mine the correlation among data from the user's interactive data, and learns the user's potential-preference expression through the deep neural network. At the same time, principal component analysis (PCA) is used to reduce the dimension of item scoring data, and calculate the similarity between item scoring data, combining the similarity between the user's potential-preference representation and the project feature representation as the final result, and recommending the project to the user [41].

2.4. Hybrid-Recommendation System Based on Heterogeneous Information Network

As a new research direction, the heterogeneous information network (HIN) has attracted extensive attention in the field of recommender systems in recent years [42]. The heterogeneous information network does not need to build a user model according to user tags. It considers multiple objects and the relationship between objects, which can fully demonstrate the complex interaction between objects in the recommendation system, and provides a new idea for personalized recommendations. Feng et al. (2012) proposed OptRank to solve the cold start problem of social recommender systems based on heterogeneous information [43]. Yu et al. (2013) further optimized the recommendation algorithm based on the relationship between heterogeneous network entities [44]. Shi et al. (2019) fused the heterogeneous information obtained by random walk with the traditional matrix-factorization model [45]. Jamali et al. (2013) combined traditional matrix-factorization algorithms with heterogeneous information networks, which are essentially context-dependent matrix-factorization models. The model considers both the general latent factors of the entity and the contextual latent factors involved in the entity [46]. These methods obtain the latent features of users and products through auxiliary information in heterogeneous information networks, but may ignore information such as historical interactions between users and products. Scholars extract meta-paths in heterogeneous information networks, and use meta-paths to measure the similarity of nodes in heterogeneous information networks to achieve recommendations. Suo (2017) combines multiple types of implicit feedback data with meta-paths for better movie recommendation [47]. Zhu (2020) used meta path theory and the DPRel correlation metric algorithm in heterogeneous information networks to calculate the correlation between authors and literature based on the construction of a disciplinary heterogeneous knowledge network model, and implemented academic literature recommendation based on the correlation ranking [48]. Niu et al. (2020) proposed a heterogeneous attention-based recurrent neural-network model. The model fuses text data and relational networks for short-term text recommendation [49]. Zhao et al. (2020) proposed a recommendation algorithm based on heterogeneous-information-network-representation learning and an attention neural network. After verification using real datasets, it was proved that this method has a better

recommendation effect than other classical algorithms [50]. Wang et al. (2020) proposed a fuzzy-recommendation algorithm based on an asymmetric heterogeneous-information network, which uses fuzzy set theory to obtain user preferences, and then calculates user similarity according to meta-paths to form recommendations [51].

To sum up, it can be seen that the current research idea based on the heterogeneous-information-network recommendation is to extract the meta-path from the heterogeneous information network and then calculate the node correlation, so as to realize the recommendation. However, in the current research on Microblog heterogeneous-information-network recommendation, the influence of users' preferences on Microblog topics and keywords on the recommendation results is not considered, and there are rich meta-paths between user nodes and topic nodes which integrate user preferences. Being able to assign higher weights to meaningful meta-paths improves recommendation accuracy. Therefore, this study proposes a Microblog text-recommendation model based on heterogeneous information networks and users' multiple trust relationships. First, by calculating the similarity and the strength of multiple trust relationships between the target user and its following users, the group of users that the target user is interested in is obtained, and the expression of the interest preference of the target user group is realized. There are multiple meta-paths between users and Microblog topics, and the author's preference is integrated to measure the correlation between users and topics, and then the similarity between users is obtained. Finally, the personalized recommendation of Microblog text is realized through the user-fusion similarity.

3. Microblog Recommendation Method Based on Meta-Path and User Trust Relationship

This chapter will introduce the research framework, the construction of the Meta-path2Vec model, the calculation of multiple trust relationships between users, and the personalized recommendation of Microblog texts.

3.1. Framework of the Research

The framework of the Microblog recommendation method based on the meta-path and user trust relationship is shown in Figure 1. It is divided into three modules, namely the Microblog data-collection and processing module, the Microblog user-similarity-calculation module and the personalized recommendation module. The Microblog user-similarity-calculation module is composed of random walk calculation based on the meta-path and multiple-trust-relationship calculation between users.

Table 1 shows the main symbols and their definitions in this paper.

Table 1. Main symbols and their definitions.

Symbols	Definition
A_t	Category of node
v_m^t	Node of specified type
$N_t(v)$	Type t neighborhood node of node v
X_v	Embedding vector of node v
$\sigma(x)$	Activation function of neural network
${v'}_k$	The k-th negative sampling node of v'
K	Negative sampling value
u_i	Node vector of user i
λ	Fusion coefficient of similarity between users
a_t^m	Topic probability distribution of Microblog a
T	Subject collection of interest of target user u_i
$topk_u$	Top k Microblog lists recommended for Microblog user u
Z_k	Normalization constant
r_{i-1}	The relevance of Microblog text with position i in the recommendation list

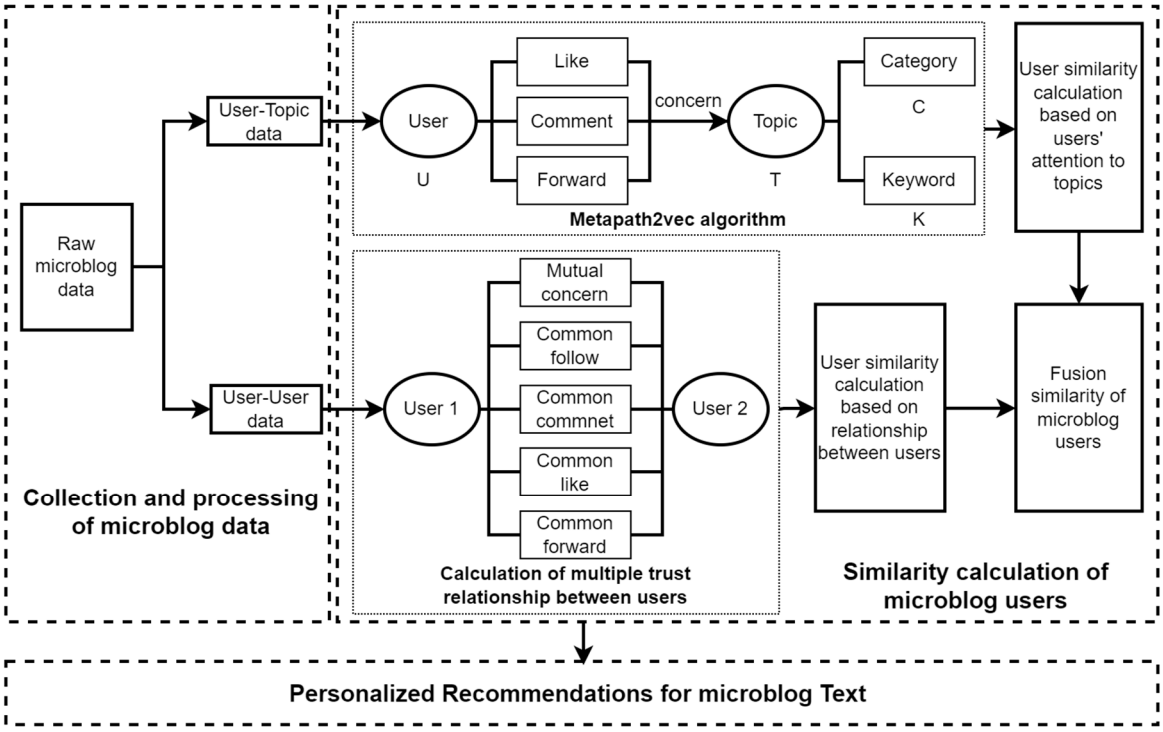

Figure 1. Framework of the research.

3.2. Construction of Metapath2Vec Model

The heterogeneous information network (HIN) is a directed graph $G = (V, E)$ with an object-type mapping function $\varphi : V \rightarrow A$ and a link-type mapping function $\phi : E \rightarrow R$. Among them, each object $v \in V$ belongs to a specific object type, denoted as $\varphi(v) \in A$, and each link $e \in E$ belongs to a specific relation type $\phi(e) \in R$; if two links belong to the same relation type, the two links have the same start object type and end-object type. When the object type $|A| > 1$, the network is called a heterogeneous information network. The heterogeneous information network structure in this paper is shown in Figure 2b.

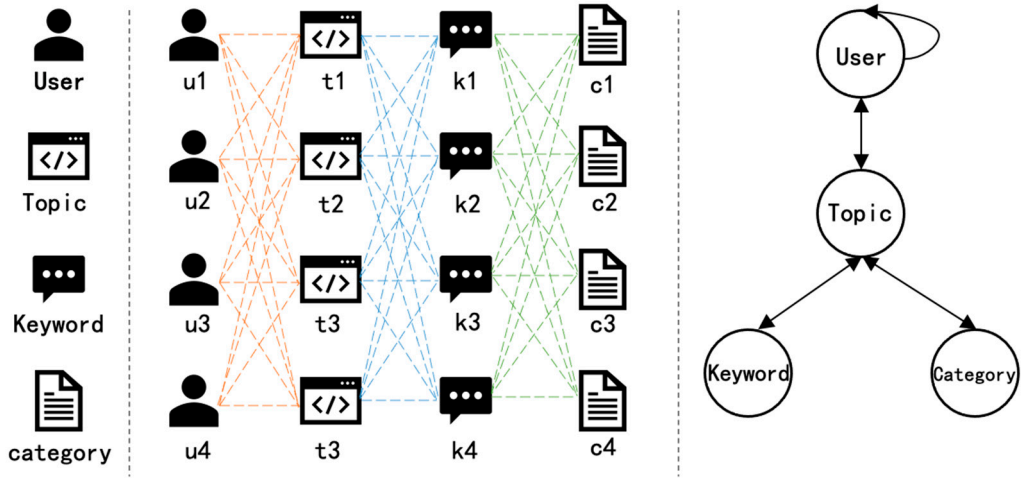

Figure 2. Heterogeneous information network and mode diagram.

3.2.1. Construction of Microblog Topic-Network Model

Sun et al. [52] used the concept of network schema to represent the meta-layer features in the network. The network schema is a meta-template of a heterogeneous information network $G = (V, E)$ with object-type mapping $\varphi : V \rightarrow A$, and link mapping $\phi : E \rightarrow R$ (G is a directed set defined on object type A and relation type set R), denoted as $T_G(A, R)$.

In this paper, the Microblog topic-network pattern $T_G(A, R)$ (as shown in Figure 2c) is the template of the Microblog topic network $G = (V, E)$, in which the knowledge nodes of the network pattern are determined by the User, Topic, Keyword and Category of topic composed of four nodes, and there are three types of edges at the same time, namely the relationship between users and topics, the relationship between topics and keywords, and the relationship between topics and category of topics.

3.2.2. Selection of Meta-Paths

The difference between the Metapath2Vec method selected in this paper and the traditional Deepwalk algorithm is that when regenerating path data, a simple random walk is no longer used, but a random walk of the meta-path is performed on the premise of manually defining the walk path. Taking Figure 2b as an example, when the number of "Keyword" nodes is much larger than other types of nodes, the path obtained by a simple random walk will collect more information from "Keyword" nodes, resulting in nodes with fewer nodes. The information is more difficult to collect, and the manual definition of the walk path provides a limit for the random walk, which solves this problem well.

Given a heterogeneous information network $G = (V, E)$, the corresponding node sequence will be generated through a random walk strategy based on the meta-path, with the meta-path MP as the constraint. The probability of random walk is shown in Formula (1).

$$P\left(v_{m+1}\middle|v_m^t, MP\right) = \begin{cases} \frac{1}{\left|N^{A_{t+1}}\left(v_m^t\right)\right|}, & \left(v_{m+1}, v_m^t\right) \in E \text{ and } \phi v_{m+1} = A_{t+1} \\ 0, & else \end{cases} \quad (1)$$

Among them, v_m^t represents the node of A_t type; $\left|N^{A_{t+1}}\left(v_m^t\right)\right|$ represents the quantity of whose node type is A_{t+1} in the neighbor nodes of node v_m^t.

Each meta-path in the network has semantic meaning, and can reflect user preferences (features). In the Microblog topic network, different user nodes can use different meta-paths to calculate the correlation between the two. The possible preferences of users in the network include Microblog topics, keywords, and topic types. The length of the meta-path cannot be set arbitrarily. If the meta-path is too long, it will increase the complexity of the calculation, and if the meta-path is too short, the semantics it contains will not be rich enough. According to the finite-behavior theorem of PathSim under infinite-length meta-paths [53], it can be inferred that infinitely increasing the meta-path length will not improve the correlation measure [53]. Therefore, the maximum length of the meta-path is limited to 4, and the meta-paths P1, P2, and P3 are selected from the Microblog topic-network model as the meta-paths used in this recommendation method, as shown in Table 2.

Table 2. Meta-paths and their meanings.

Meta-Path	Meaning of Meta-Paths
P1 = UTU	Users who follow the same topic as the selected user
P2 = UTKTU	Users who follow a topic with the same keywords as a selected user's topic of interest
P3 = UTCTU	Users who follow topics of the same type as a selected user's topic

3.2.3. Generation of Node Representation Vectors

Through the random walk of the fixed path, the neighborhood sequence of each node is obtained. We need to use the heterogeneous skip-gram model to obtain the embedding vector of each node to form the node-embedding matrix **X**.

For a heterogeneous information network $G = (V, E)$, the network mode is $T_G(A, R)$; by maximizing the conditional probability of the heterogeneous node neighborhood $N_t(v)$,

$t \in A$ under the condition of a given node v, the vector representation of the effective nodes of the heterogeneous information network G is obtained, as shown in Formula (2).

$$\underset{\theta}{\operatorname{argmax}} \sum_{v \in V} \sum_{t \in A} \sum_{v' \in N_t(v)} logp(v'|v;\theta) \tag{2}$$

Among them,

$$p(v'|v;\theta) = \frac{e^{X_{v'} \cdot X_v}}{\sum_{u \in V} e^{X_u \cdot X_v}} \tag{3}$$

In the formula, $N_t(v)$ represents the t-th type of neighbor node of node v.

In order to effectively optimize the objective function and improve the training efficiency, this paper adopts the method proposed by Mikolov et al. [54], which introduces negative sampling, randomly samples a group of nodes not belonging to the neighborhood nodes from the network as a group of training data, and establishes a two-class model to replace the function of softmax; the same negative sampling is also used in metapath2vec. Given a negative sampling data set M, define:

$$logp(v'|v;\theta) = log\sigma(\boldsymbol{X}_{v'} \cdot \boldsymbol{X}_v) + \sum_{m=1}^{M} \log \sigma(-\boldsymbol{X}_{v'_k} \cdot \boldsymbol{X}_v) \tag{4}$$

where $\boldsymbol{X}_v$ represents the embedding vector of node v, $\sigma(x) = \frac{1}{1+exp(-x)}$ is the activation function of the neural network. v'_k is the kth negative sampling node of v', and K is the number of negative samples.

3.2.4. Similarity Calculation Based on Users' Attention to Topics

This paper designs three different meta-paths, namely the path based on topic information (P1), the path based on topic-keyword information (P2) and the path based on topic-type information (P3). This paper will integrate multiple paths to comprehensively consider the user's information preferences, and provide users with personalized Microblog content recommendations. Therefore, there are two cases for the calculation of user similarity, namely, calculating the user similarity based on a single meta-path and calculating the comprehensive user similarity by fusing multiple meta-paths.

User similarity is calculated based on a single meta-path. Through the embedding technology of the skip-gram model, we have completed the generation of the representation vector of each node of the meta-path. The smaller the distance between two node vectors in space, the higher the similarity between the two. Therefore, the similarity between users can be converted into the calculation of the spatial distance between two user node vectors. The distance between two user node vectors is inversely proportional to the similarity between the two users. By calculating the similarity between users, it can be used for users with high similarity. Providing the corresponding Microblog text-recommendation list realizes the personalized recommendation of Microblog text. The similarity between users is determined by the magnitude and direction of the user node vector. The vector quantity-product formula can consider the magnitude and direction of the vector, so this paper uses the quantity product to calculate the distance of nodes. Specifically, given the query user nodes u_i and u_j, the similarity calculation formula between them is shown in Formula (5).

$$sim(\boldsymbol{u}_i, \boldsymbol{u}_j) = e^{-\boldsymbol{u}_i \cdot \boldsymbol{u}_j} \tag{5}$$

Among them, $\boldsymbol{u}_i$ and $\boldsymbol{u}_j$ are the node vectors of the user.

Integrate multiple meta-paths to calculate the comprehensive similarity of users. When there are different paths between two users, the similarity between the two under each path is calculated separately, and then the comprehensive similarity of the two users is obtained by adding them together.

3.3. Calculation of Multiple Trust Relationships between Users

The calculation of the trust relationship between users is mainly to calculate the trust relationship between users through social behaviors such as mutual attention, common attention, common comment, common like, and common forwarding among users. The strength of the comprehensive trust relationship between users is calculated.

(1) Calculation of trust strength of mutual-attention relationship between users

In the platform of Microblog, the main way for the dissemination of Microblogging information is the relationship between users' attention and being followed. Obtaining Microblogging information through the users' Microblogging friends is one of the main sources of information for users. A relationship is an explicit and direct trust relationship [55]. Suppose user ***i*** pays attention to user ***a***, user ***b***, and user ***j***, indicating that user ***i*** and the other three users may have similar preferences. Users ***a***, ***b***, and ***j*** may have a positive impact on user ***i***'s preferences. There is a certain trust relationship. If, among the four users, only user ***j*** and user ***i*** follow each other, it means that the trust relationship between the two is higher than that of the other two users mentioned. When there are frequent interactive behaviors such as mutual likes, mutual comments, and mutual reposts, it indicates that the trust relationship between the two users is stronger, and the similarity of Microblog-article preferences is higher. The calculation method for the trust strength of the mutual-attention relationship between users is shown in Formula (6).

$$T_{ij}^{1} = \frac{N_{ij}}{N_i} \tag{6}$$

Among them, N_{ij} represents the total number of interactive behaviors between user ***i*** and user ***j***, and N_i represents the total number of behaviors such as commenting, reposting, and liking between user ***i*** and user ***i*** on the Microblog platform.

(2) Calculation of trust strength of common concern relationship among users

Suppose user ***i*** follows user ***a***, and user ***j*** also follows user ***a***, that is, the two users follow at least one other user in common. At this time, there is a common-follow relationship between user ***i*** and user ***j***, and there is a certain relationship between the two users, a trust relationship. In the Microblogging platform, users with high similarity of interests and preferences are more likely to form a common concern relationship, and the more users that two users follow in common, the higher the strength of the trust relationship between the two users. The calculation method of the trust strength of the common-concern relationship between users is defined as shown in Equation (7).

$$T_{ij}^{2} = \frac{num[F(i) \cap F(j)]}{num[F(i) \cup F(j)]} \tag{7}$$

Among them, $F(i)$ represents the list of all users followed by user ***i***.

(3) Calculation of trust strength of common comment relationship between users.

It is assumed that both user ***i*** and user ***j*** have commented on the Microblog content published by user ***a*** at the same time, indicating that the two users have a common comment relationship, and their interests and preferences may have a certain similarity. When the two users commented on more Microblog content, the more similar their interests and preferences in the Microblog content, and the higher the strength of the trust relationship between the two users. The calculation method of the trust strength of the common comment relationship between users is defined as shown in Equation (8).

$$T_{ij}^{3} = \frac{num[C(i) \cap C(j)]}{num[C(i) \cup C(j)]} \tag{8}$$

Among them, $C(i)$ represents the list of all Microblogs commented by user ***i***.

(4) Calculation of trust strength of common like relationship among users

On the platform of Microblog, users can simply express their attitude towards blog posts published by other users by liking them. When a user likes a blog post, it can express that the user has "read" the message, or express his attitude of "approval" or "opposition" to the opinions expressed by the blogger, but, in the final analysis, it is all about the blog post, a kind of "attention". Assuming that both user *i* and user *j* like the Microblog content published by user ***a*** at the same time, it means that the two users have a common like relationship. The more Microblog liked by the two users, the higher the similarity of interests and preferences between the two users, and the higher the strength of the trust relationship between the two. The calculation method of the trust strength of the common like relationship between users is defined as shown in Formula (9).

$$T_{ij}^{4} = \frac{num[L(i) \cap L(j)]}{num[L(i) \cup L(j)]} \tag{9}$$

Among them, $L(i)$ represents the list of all Microblogs liked by user ***i***.

(5) Calculation of trust strength of mutual forwarding relationship between users

Assuming that both user ***i*** and user ***j*** have forwarded a certain Microblog content, then the two users have a common forwarding relationship. The greater the number of Microblog content shared by the two, the higher the similarity of their interests and preferences in the Microblog content, and the higher the strength of the trust relationship. The calculation method of the trust strength of the common like relationship between users is defined as shown in Formula (10).

$$T_{ij}^{5} = \frac{num[R(i) \cap R(j)]}{num[R(i) \cup R(j)]} \tag{10}$$

Among them, $R(i)$ represents the list of all Microblogs forwarded by user ***i***.

(6) Calculation of fusion strength of multiple trust relationships among users

By means of linear weighting, Equations (1)–(5) are fused to calculate the strength of multiple trust relationships among Microblog users, as shown in Equation (11).

$$T_{ij} = \alpha T_{ij}^{1} + \beta T_{ij}^{2} + \gamma T_{ij}^{3} + \delta T_{ij}^{4} + \varepsilon T_{ij}^{5} \tag{11}$$

Among them, $\alpha, \beta, \gamma, \delta, \varepsilon$ are weight coefficients, $\alpha + \beta + \gamma + \delta + \varepsilon = 1$ In this paper, $\alpha = \beta = \gamma = \delta = \varepsilon = 0.2$.

3.4. Personalized Recommendation of Microblog Text

According to the user similarity $sim(u_i, u_j)$ of the user's attention to the topic and the user similarity T_{ij} of the attention relationship between users, the fusion similarity between user ***i*** and user ***j*** is obtained by linear weighting, as shown in Formula (12).

$$R(u_i, u_j) = \lambda * sim(u_i, u_j) + (1 - \lambda) T_{ij} \tag{12}$$

Among them, λ is the weight coefficient; the calculated fusion similarity is sorted in descending order, and TOP-N similar users are selected as the interest group $\boldsymbol{Q}$ of the target user u_i, and the group-interest distribution of the target user u_i on the Microblog topic is calculated as shown in Formula (13).

$$I_t^{Q} = \sum_{j \in Q} R(u_i, u_j) * u_{jt}^{m} \tag{13}$$

where u_{jt}^{m} represents the topic-probability distribution of user u_j.

Assuming that the Microblog text collection of similar users of user u_i in the time window $\boldsymbol{W}$ generates attention behaviors such as A_{txt}, the LDA model is used to perform

topic mining on each Microblog a_{txt} in the collection, and its topic-probability distribution a_t^m is obtained, and the pair of user u_i is calculated. The topic-interest degree of the new Microblog text is shown in Equation (14).

$$I(u_i, a_{txt}) = \sum_{t \in T} (a_t^m * I_t^Q) \tag{14}$$

where $\boldsymbol{T}$ represents the set of topics of interest of the target user u_i. Sort the calculation results of $I(u_i, a_{txt})$ in descending order, and recommend the top-ranked TOP-N Microblog texts to the target user u_i, so as to realize the personalized recommendation of the Microblog texts.

4. Experimental Results and Analysis

In this section, we will sequentially introduce data sources and preprocessing for empirical analysis, comparison algorithms and evaluation indicators, parameter selection, experimental results and experimental-result analysis.

4.1. Data Source and Preprocessing

This paper filters the definition of microblog content category under the "popular microblog classification" option in the microblog platform. Select 10 categories of "technology", "finance", "star", "sports", "emotion", "funny", "photography", "tourism", "beauty" and "food" as the "category" node of this paper. According to different categories, there are 217 users who crawled Weibo text and customer information and generated likes, comments, forwarding and other attention behaviors under the 10 Weibo content, with the highest popularity in the above categories. Use the Python tool to write a crawler program to crawl the following user ID, fan ID, published, commented and forwarded Weibo text content of the above users, and obtain 33,245 relevant data.

Before the empirical analysis, the data needs to be preprocessed. Since there are a lot of non-text content such as pictures and expressions on Microblog, when the crawler is used for information crawling, only the HTML information of the pictures or expressions will be crawled. This kind of data is removed, and at the same time, the links in the blog post and the information of @other Microblog users are removed, and 32,019 pieces of data are obtained after preliminary cleaning and deduplication. The "jieba" toolkit in the Python software can segment sentences more accurately in full mode, and can be competent for word segmentation in text processing. The text data is segmented, and the stop words are removed from the segmented text data based on the stop-word list of the Harbin Institute of Technology. Using the LDA model to extract topics from the Microblog texts that different users have followed, the "topic" nodes and "keywords" nodes are obtained. We collect each user's Microblog content attention, randomly select 80% of each user's Microblog content attention as the training set, and the remaining 20% as the test set.

4.2. Comparison Algorithm and Evaluation Index

When selecting parameters, this paper selects the mean absolute error (*MAE*) and the root-mean-square error (*RMSE*) as the error-evaluation-value indicators. This is shown in Formulas (15) and (16).

$$MAE = \frac{1}{n} \sum_{i=1}^{n} |y_i - \hat{y}_i| \tag{15}$$

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2} \tag{16}$$

When evaluating the recommendation effect, this paper selects the precision rate (*Precision@k*), recall rate (*Recall@k*), *F*1 value (*F1@K*), and normalized depreciation cumulative

gain (*NDCG@k*) of the first *k* results as an evaluation index. The index calculation method is shown in Formulas (17)–(20).

$$Precision@k = \frac{topk_u \cap D_{test_u}}{topk_u} \tag{17}$$

$$Recall@k = \frac{topk_u \cap D_{test_u}}{D_{test_u}} \tag{18}$$

$$F1@K = \frac{2 * Precision@k * Recall@k}{Precision@k + Recall@k} \tag{19}$$

$$NDCG@K = Z_k \sum_{i=1}^{K} \frac{2^{r_{i-1}}}{log_2(i+1)} \tag{20}$$

where $topk_u$ represents the list of the top $\boldsymbol{k}$ Microblogs recommended by Microblog user $\boldsymbol{u}$, D_{test_u} represents the set of Microblog texts in the test set that user $\boldsymbol{u}$ generates for attention behaviors, Z_k is the normalization coefficient, and r_{i-1} is the relevance of the Microblog text at position $\boldsymbol{i}$ in the recommendation list.

In this paper, four common recommendation algorithms are selected for comparison with the algorithm proposed in this paper.

(1) MF [56]. Matrix factorization is a recommendation model based on the idea of collaborative filtering. The model takes only the user-item rating matrix as input, and then optimizes to obtain two low-rank matrices to predict unknown ratings.
(2) PMF [56]. Probabilistic matrix factorization is a traditional rating-prediction model that assumes a Gaussian distribution for the latent vectors of users and items, and then performs matrix factorization.
(3) Deepwalk [57]. In this method, the Skip-gram algorithm is applied to the graph network for the first time, and the node pairs in the k-hop field are obtained by uniform random walk on the homogeneous network to form the node sequence, and then the Skip-gram algorithm is used to learn the node representation. The experiment does not distinguish between user nodes and item nodes. After obtaining the representation of the node, the user node and the item node are used as the input of MLP, and the recommendation is made according to the prediction score.
(4) Node2vecp [58]. This method is improved on the basis of the Deepwalk model. It combines the BFS and DFS strategies during random walk, defines parameters $\boldsymbol{p}$ and $\boldsymbol{q}$ to initialize the probability transition matrix, and walks according to the probability of obtaining a fixed-length node sequence, and then learns the low-dimensional embedded representation of the node. The recommended method after obtaining the node representation is the same as above.

4.3. Selection of Parameter

This section analyzes the influence of the word vector dimension and the number of iterations on the recommendation results of the Metapath2Vec model, and selects the most suitable parameters as the parameters of the fusion model. At this time, the weight coefficient λ is set to be 1, which means that only the influence of the Metapath2vec model on the recommendation results of the Microblog text is considered.

4.3.1. Selection of Word Vector Dimensions

Experiments are performed by setting multiple different word vector dimensions, as shown in Figure 3. As can be seen from Figure 3, the data set selected in this paper is more sensitive to dimensional changes. As the vector dimension increases from 8 to 256, the MAE value and RMSE value of the model first decrease and then increase. When the word vector dimension is 128, the MAE value and RMSE value of the Metapath2vec model are the smallest, so the dimension of the word vector model is set to 128.

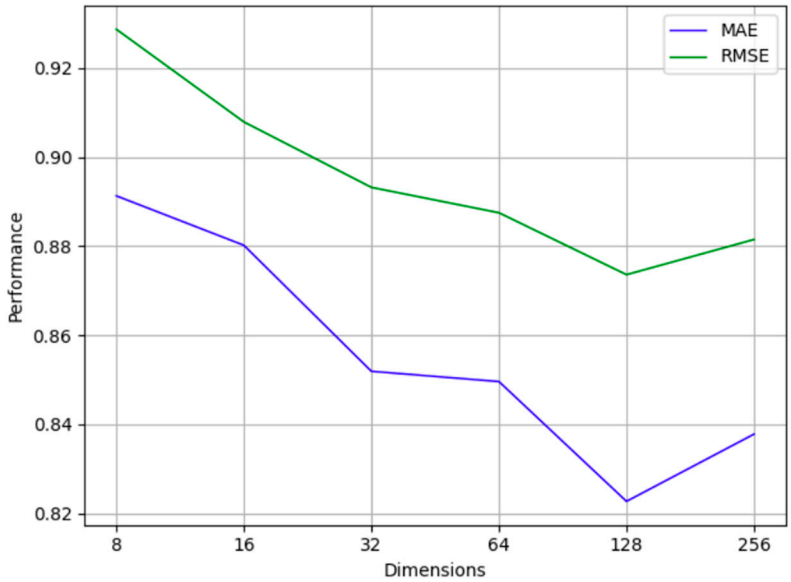

Figure 3. Experimental results of the model under different word vector dimensions.

4.3.2. Selection of the Number of Iterations

For different datasets and different application tasks, the number of iterations is set differently, so this parameter is experimentally analyzed, and the results are shown in Figure 4. As can be seen from Figure 4, with the increase in the number of iterations, the error of the algorithm recommendation gradually decreases, and becomes stable after 250. This shows that the increase in the number of iterations within a certain range (set to 400 in this paper) can effectively use the information in the data to improve the fitting ability of the algorithm; however, over-fitting is prone to occur after the range is exceeded, resulting in poor experimental results.

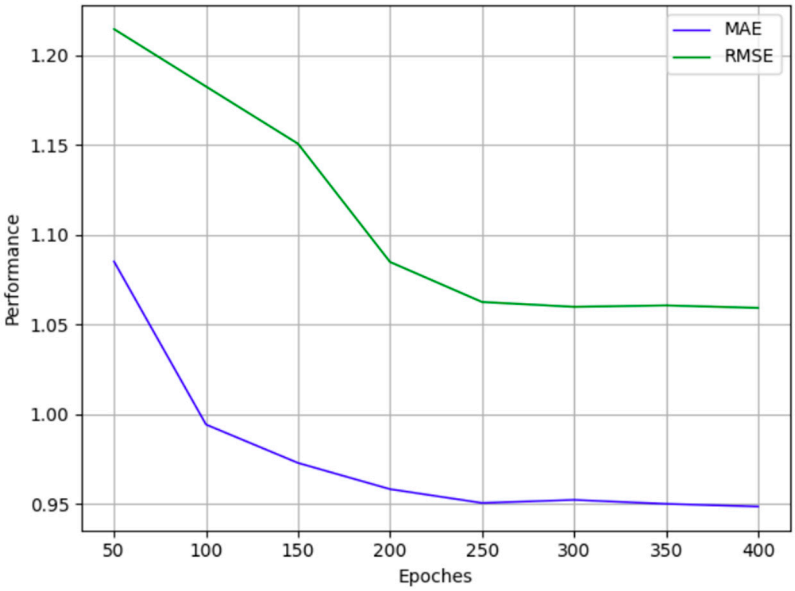

Figure 4. Implementation results of the model under different iterations.

4.4. Analysis of Experimental Results

The comparison results of different meta-paths are shown in Figure 5. Through the comparative-analysis results, we draw the following conclusions:

(1) From a global perspective, the fusion-path method is better than other single paths in the results of each indicator, which shows that the fusion path can better combine the

content, topic type and keyword information of the Microblog text to capture more information comprehensively.

(2) From the comparison of the recommendation results of the single paths P1, P2 and P3, except for individual results, in most cases, the recommendation results of the path P2 based on topic keywords and the path P3 based on topic categories are better than the baseline path P1, that is to say, in general, the keywords and categories of topics can effectively mine users' demand preferences for Microblog texts to a certain extent, and improve the accuracy of the recommendation results.

(3) When k = 20, the precision rate, recall rate and F1 value under different paths all achieve the maximum value, indicating that when the number of Microblogs in the Microblog recommendation list is 20, the recommendation effect is the best.

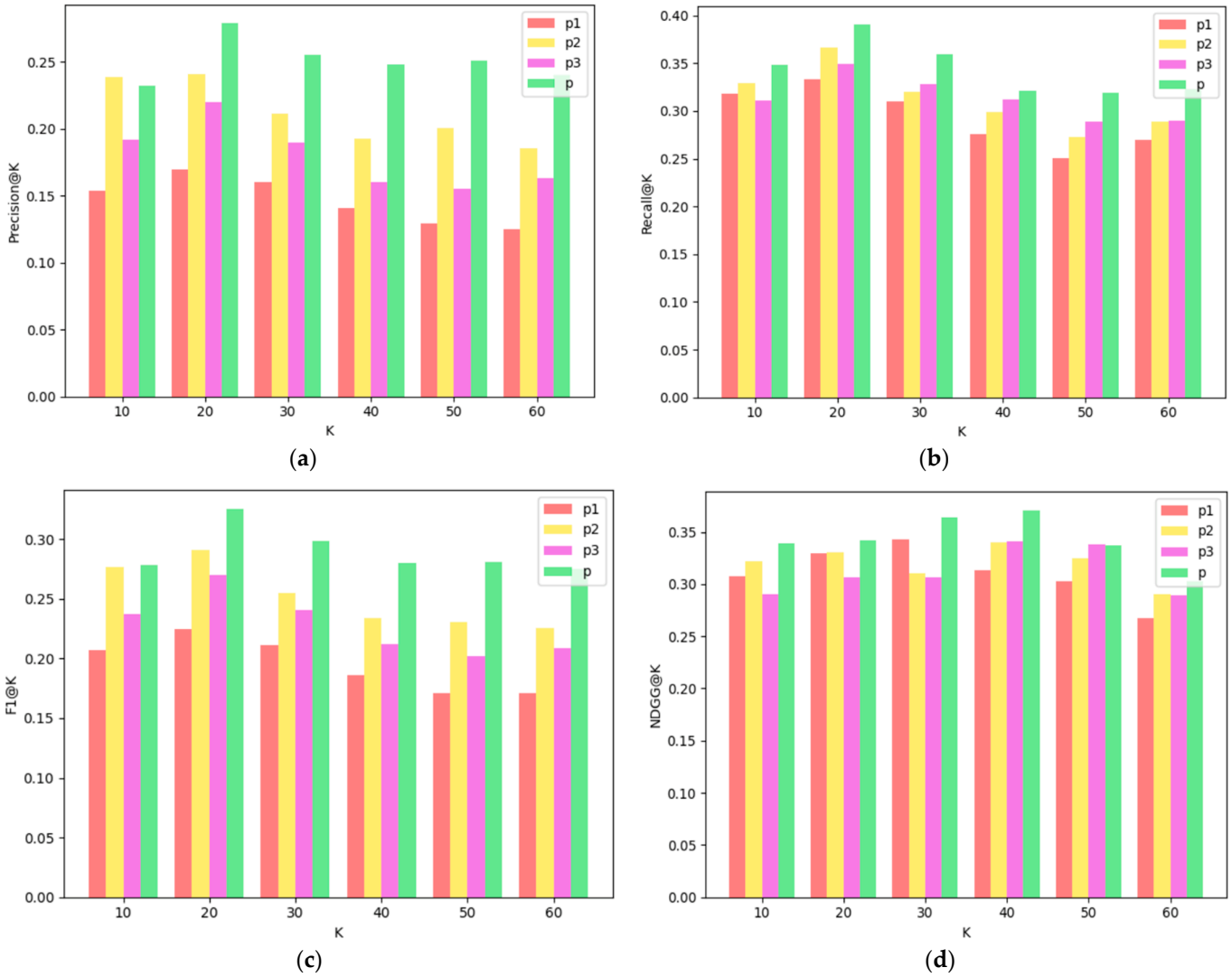

Figure 5. (**a**) Comparison of precision; (**b**) Comparison of recalls under different paths; (**c**) Comparison of F1 values; (**d**) Comparison of NDCG values under different paths.

For the determination of the value of λ, according to the results obtained in Figure 5, set k = 20, and calculate the recommended effect of the model under 10 values of λ in the [0,1] numerical interval with an interval of 0.1. The result is shown in Figure 6.

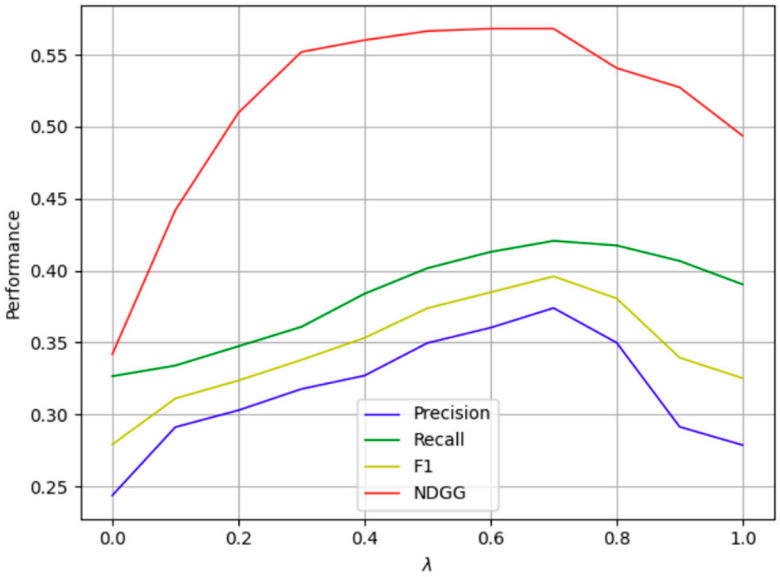

Figure 6. Recommendation results under different λ.

In Figure 6, when $\lambda = 0$, it means that a single trust relationship is used at this time to calculate the similarity between users. When $\lambda = 1$, it means that a single random walk based on the meta-path is used at this time. Taking the recommendation algorithm that calculates the similarity of users, we can see that in most cases, the recommendation effect of the fusion algorithm is better than that of the single algorithm. When two single algorithms are compared, it is obvious that the recommendation effect of the meta-path random-walk algorithm is better. It is better than the recommendation effect of the multiple-trust-relationship calculation. When $\lambda = 0.7$, the precision rate, recall rate, F1 value and NDCG value of the recommended results all achieve the maximum value, and the recommendation effect is the best. Therefore, this paper takes $\lambda = 0.7$, and the user-fusion similarity-calculation formula is obtained as shown in Formula (21).

$$R(u_i, u_j) = 0.7 * sim(u_i, u_j) + 0.3T_{ij} \tag{21}$$

Table 3 shows the comparison results of the recommendation algorithm proposed in this paper with the fusion of the heterogeneous information network and multiple trust relationships and other models. In Table 3, the data in bold are the best results. By comparing the results, we draw the following conclusions:

(1) On the whole, by comparing the average value of the evaluation indicators, it can be found that the algorithm proposed in this paper shows the best results in the four evaluation indicators.

(2) Compared with the Deepwalk model and the Node2vec model, the average precision of the method in this paper is 0.3211, which is about 0.2 and 0.07 higher than the Deepwalk and Node2vec algorithms, respectively, which are both random-walk-related algorithms. The effect of the method proposed in this paper is better, indicating that the trust relationship between fusion users has a positive impact on the recommendation results.

(3) From the three indicators of precision rate, recall rate and F1 value, when k = 20, the model in this paper has the best effect. In the actual situation, the number of user-focused Microblogs is limited, and the excessive number of Microblog texts in the recommendation list will reduce the accuracy of the recommendation, and the visualization method is not easy to view. When the number of Microblog text recommendations is too small, it may not be possible to cover users' demands and preferences as much as possible. Therefore, according to the calculation results in this paper, in order to ensure the accuracy of the recommendation results, we choose to

recommend 20 candidate Microblog-text lists for the user, which can satisfy the user's demand preference to the greatest extent.

Table 3. Comparison of recommendation results of different algorithms.

k	*Precision@k*					*Recall@k*				
	MF	**PMF**	**Deepwalk**	**Node2vec**	**Ours**	**MF**	**PMF**	**Deepwalk**	**Node2vec**	**Ours**
10	0.0936	0.1205	0.0948	0.2320	**0.3676**	0.2901	0.3691	0.2376	0.3488	**0.3974**
20	0.0915	0.0963	0.1074	0.2786	**0.3739**	0.2677	0.3704	0.2419	0.3903	**0.4206**
30	0.0742	0.0805	0.1006	0.2549	**0.3403**	0.2372	0.3958	0.273	0.3596	**0.4083**
40	0.0537	0.0632	0.0941	0.2481	**0.3168**	0.2196	0.3641	0.2983	0.3212	**0.3802**
50	0.0396	0.0596	0.0733	0.2510	**0.2907**	0.2069	0.3216	0.2433	0.3187	**0.3473**
60	0.0248	0.0401	0.0603	0.2400	**0.2432**	0.1603	0.2804	0.2302	0.3232	**0.3245**
Average	0.0629	0.0767	0.0884	0.2508	**0.3221**	0.2303	0.3502	0.2541	0.3436	**0.3797**
k	***F1@k***					***NDCG@k***				
	MF	**PMF**	**Deepwalk**	**Node2vec**	**Ours**	**MF**	**PMF**	**Deepwalk**	**Node2vec**	**Ours**
10	0.1415	0.1817	0.1355	0.2786	**0.3819**	0.2337	0.2904	0.0936	0.3389	**0.5473**
20	0.1364	0.1529	0.1488	0.3251	**0.3959**	0.2418	0.3065	0.0718	0.3417	**0.5519**
30	0.113	0.1338	0.147	0.2983	**0.3712**	0.2602	0.3196	0.0606	0.3639	**0.5794**
40	0.0863	0.1077	0.1431	0.28	**0.3456**	0.2698	0.3318	0.0415	0.3702	**0.5838**
50	0.0665	0.1006	0.1127	0.2808	**0.3165**	0.2397	0.3037	0.0377	0.3368	**0.5506**
60	0.043	0.0702	0.0956	0.2754	**0.278**	0.2003	0.2836	0.0294	0.3024	**0.5293**
Average	0.0978	0.1245	0.1305	0.2897	**0.3482**	0.2409	0.3059	0.0558	0.3423	**0.5571**

5. Conclusions

This paper proposes a Microblog-text-recommendation method that integrates heterogeneous-information-network theory and users' multiple trust relationships. Firstly, according to the characteristics of the Microblog text, the "user" node, "type" node, "topic" node and "keyword" node are generated respectively, and then the meta-paths with the walking rules are set to travel between nodes, combined with embedding technology which learns the node vector, calculates the spatial distance between user nodes to obtain the user similarity based on the meta-path random walk, and then fuses it with the calculated user similarity based on multiple user trust relationships to obtain user interest groups. The group obtains the collection of Microblog texts that users may be interested in, and realizes the personalized recommendation of Microblog texts. The experimental results show that the proposed method for Microblog text recommendation by fusing heterogeneous information networks and users' multiple trust relationships is feasible and effective. The method proposed in this paper explores the feasibility of mapping heterogeneous information networks to Microblog text, and provides ideas for the follow-up development of Microblog text recommendation. However, from the results of Microblog text recommendation, although the indicators of the method proposed in this paper are better than other comparison algorithms, the recommendation performance still needs to be improved, and the follow-up work will improve the performance of the recommendation method.

Author Contributions: Conceptualization, C.Y.; methodology, L.L.; software, L.L.; validation, L.L.; formal analysis, L.L.; investigation, L.L.; resources, L.L.; data curation, L.L.; writing—original draft preparation, L.L.; writing—review and editing, L.L.; visualization, L.L.; supervision, C.Y.; project administration, C.Y.; funding acquisition, C.Y. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by Natural Science Foundation of Shandong Province [No. ZR2022MG059]. This research was supported by Social Science Planning Research Program of Qingdao [No. QDSKL2201131]. In addition, this research was supported by National Bureau of Statistics, National Statistical Science Research Project Key Project [No. 2022LZ31].

Data Availability Statement: The data used for this study will be made available upon request to the authors.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Li, C.Q.; Wu, F.; Sun, Q. Research on information spreading effects in the social interaction context. *J. Manag. Eng.* **2023**, *37*, 22–34.
2. Zhou, J.J.; Zuo, M.Y. Research on the relationship between online and offline interaction, group differentiation and knowledge sharing—Empirical analysis based on virtual communities. *China Manag. Sci.* **2012**, *20*, 185–192.
3. Bao, X.; Wang, M.R.; Liu, G.F. A novel text classification mathod based on topic model and transfer learning. *J. Shandong Univ. Sci. Technol. Nat. Sci.* **2021**, *40*, 80–88.
4. Jiao, Y.Y.; Li, Z.Z.; Shen, Z.F. Research on the diffusion mechanism of new products in the context of social networks: Generation process based on peer influence. *J. Manag. Eng.* **2020**, *34*, 105–113.
5. Lin, J.; Yang, Z.J. User's network behavior characteristics and professional knowledge level—An empirical study based on registered users of "Auto Home". *Manag. Rev.* **2021**, *33*, 331–340.
6. Li, J.P.; Cao, N.; Zhang, Q.; Zhang, W.P.; Ji, S.J. Online social network groups discovery algorithm considering themes and time. *J. Shandong Univ. Sci. Technol. Nat. Sci.* **2021**, *40*, 95–102.
7. Ding, Y.; Liu, J.; Jiang, C.Q.; Liang, C.Y. A study of friends recommendation algorithm considering users' preference of making friends in the LBSN. *Syst. Eng. Theory Pract.* **2017**, *37*, 2975–2982.
8. Baeza-Yates, R.; Ribeiro-Neto, B. *Modern Information Retrieval*; ACM Press: New York, NY, USA, 1999.
9. Belkin, N.J.; Croft, W.B. Information filtering and information retrieval: Two sides of the same coin? *Commun. ACM* **1992**, *35*, 29–38. [CrossRef]
10. Salton, G. *Automatic Text Processing: The Transformation, Analysis, and Retrieval of Information by Computer*; Addison-Wesley: Reading, MA, USA, 1989; pp. 169–182.
11. Lang, K. Newsweeder: Learning to filter netnews. In Proceedings of the Twelfth International Conference on Machine Learning, Tahoe City, CA, USA, 9–12 July 1995; Morgan Kaufmann: Burlington, MA, USA, 1995; pp. 331–339.
12. Mooney, R.J.; Roy, L. Content-based book recommending using learning for text categorization. In Proceedings of the Fifth ACM Conference on Digital Libraries, San Antonio, TX, USA, 2–7 June 2000; pp. 195–204.
13. Pazzani, M.; Billsus, D. Learning and revising user profiles: The identification of interesting web sites. *Mach. Learn.* **1997**, *27*, 313–331. [CrossRef]
14. Li, L.; Chu, W.; Langford, J.; Schapire, R.E. A Contextual bandit Approach to Personalized News Article recommendation. In Proceedings of the 19th International Conference on World Wide Web, Raleigh, NC, USA, 26–30 April 2010; ACM Press: New York, NY, USA, 2010; pp. 661–670.
15. Jiang, S.H.; Xue, F.L. A content-based recommendation method based on collaborative filtering prediction and fuzzy similarity improvement. *Data Anal. Knowl. Discov.* **2014**, *30*, 41–47.
16. Lei, K.; Liu, S.B.; Li, D. Content-based point of interest recommendation under real-time traffic conditions. *Comput. Eng.* **2017**, *43*, 147–152.
17. Su, X.; Khoshgoftaar, T.M. A survey of collaborative filtering techniques. *Adv. Artif. Intell.* **2009**, *2009*, 109–118. [CrossRef]
18. Billsus, D.; Pazzani, M.J. Learning Collaborative Information Filters. In Proceedings of the Fifteenth International Conference on Machine Learning, Madison, WI, USA, 24–27 July 1998; Morgan Kaufmann Publishers Inc.: San Francisco, CA, USA, 1998; Volume 98, pp. 46–54.
19. Marlin, B.M. Modeling user rating profiles for collaborative filtering. In *Advances in Neural Information Processing Systems*; MIT Press: Cambridge, MA, USA, 2004; pp. 627–634.
20. Pavlov, D.Y.; Pennock, D.M. A maximum entropy approach to collaborative filtering in dynamic, sparse, high-dimensional domains. In *Advances in Neural Information Processing Systems*; MIT Press: Cambridge, MA, USA, 2003; pp. 1465–1472.
21. Pearson, K.L., III. On lines and planes of closest fit to systems of points in space. *Lond. Edinb. Dublin Philos. Mag. J. Sci.* **1901**, *2*, 559–572. [CrossRef]
22. Ungar, L.H.; Foster, D.P. Clustering methods for collaborative filtering. *AAAI Workshop Recomm. Syst.* **1998**, *1*, 114–129.
23. Hofmann, T. Collaborative filtering via gaussian probabilistic latent semantic analysis. In Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Toronto, ON, Canada, 28 July–1 August 2003; ACM: New York, NY, USA, 2003; pp. 259–266.
24. Y, X.; He, Y.D.; Liang, H.T.; Jiang, F.; Du, J.W. A Top-k crowdsourcing developer recommendation method considering interest preference. *J. Shandong Univ. Sci. Technol. Nat. Sci.* **2021**, *40*, 58–70.
25. Dwork, C. Differential privacy. In Proceedings of the 33rd International Colloquium on Automata, Languages and Programming, Venice, Italy, 10–14 July 2006; Springer: Berlin, Germany, 2006; pp. 1–12.
26. Friedman, A.; Schuster, A. Data mining with differential privacy. In Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Washington, DC, USA, 24–28 July 2010; pp. 493–502.
27. Piao, C.; Shi, Y.; Yan, J.; Zhang, C.; Liu, L. Privacy-preserving governmental data publishing: A fog-computing-based differential privacy approach. *Future Gener. Comput. Syst.* **2019**, *90*, 158–174. [CrossRef]

28. Hou, J.; Li, Q.; Cui, S.; Meng, S.; Zhang, S.; Ni, Z.; Tian, Y. Low-cohesion differential privacy protection for industrial internet. *J. Supercomput.* **2020**, *76*, 8450–8472. [CrossRef]
29. McSherry, F.; Mironov, I. Differentially private recommender systems: Building privacy into the netflix prize contenders. In Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Paris, France, 28 June–1 July 2009; pp. 627–636.
30. He, M.; Chang, M.M.; Wu, X.F. A collaborative filtering recommendation method based on differential privacy protection. *Comput. Res. Dev.* **2017**, *54*, 1439–1451.
31. Goldberg, K.; Roeder, T.; Gupta, D.; Perkins, C. Eigentaste: A constant time collaborative filtering algorithm. *Inf. Retr.* **2001**, *4*, 133–151. [CrossRef]
32. Gao, Y.; Wang, X.; Guo, L.; Chen, Z.M. Learing to recommend with collaborative matrix factorization for new users. *J. Comput. Res. Dev.* **2017**, *54*, 1813–1823.
33. Liu, G.S.; Su, B. Collaborative filtering recommendation algorithm combining community structure and personal interests. *Comput. Eng. Des.* **2018**, *39*, 3420–3424.
34. Wei, L.; Guo, X.Y. Personalized Recommendation Model of Knowledge Payment Platform Combining User Portrait and Collaborative Filtering. *Inf. Stud. Theory Appl.* **2021**, *44*, 188–193.
35. Zhang, J. LA fusion collaborative filtering algorithm based on rating information entropy. *J. Nanjing Univ. Posts Telecommun.* **2021**, *44*, 76–81.
36. Yang, X.W.; Guo, Y.; Liu, Y.; Steck, H. A Survey of Collaborative Filtering Based Social Recommender Systems. *Comput. Commun.* **2014**, *41*, 2–10. [CrossRef]
37. Bao, J.; Zheng, Y.; Wilkie, D.; Mokbel, M. Recommendations in Location-Based Social Networks: A Survey. *GeoInformatica* **2015**, *19*, 525–565. [CrossRef]
38. Guo, Q.Y.; Zhuang, F.Z.; Qin, C.; Zhu, H.; Xie, X.; Xiong, H.; He, Q. A Survey on Knowledge Graph-Based Recommender Systems. *IEEE Trans. Knowl. Data Eng.* **2020**, *34*, 3549–3568. [CrossRef]
39. Zhang, X.M.; Jiang, S.Y.; Li, X. Hybrid recommendation algorithm based on network and tag. *Comput. Eng. Appl.* **2015**, *51*, 119–124.
40. Yang, S.; Wang, J. Improved hybrid recommendation algorithm based on stack noise reduction self-coder. *Comput. Appl.* **2018**, *38*, 42–47.
41. Yuan, W.W.; Peng, D.L.; Wu, S.H. Hybrid recommendation algorithm supported by self-attention mechanism. *Micro Comput. Syst.* **2019**, *7*, 26–31.
42. Zhao, C.; Zhang, K.H.; Liang, J.Y. Asymmetric Recommendation Algorithm in Heterogeneous Information Network. *J. Front. Comput. Sci. Technol.* **2020**, *14*, 939–946.
43. Feng, W.; Wang, J.Y. Incorporating heterogeneous information for personalized tag recommendation in social tagging systems. In Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Beijing, China, 12–16 August 2012; pp. 1276–1284.
44. Yu, X.; Ren, X.; Gu, Q.Q.; Sun, Y.; Han, J. Collaborative filtering with entity similarity regularization in heterogeneous information networks. In Proceedings of the IJCAI-13 Workshop on Heterogeneous Information Network Analysis, Beijing, China, 5 August 2013; p. 27.
45. Shi, C.; Hu, B.B.; Zhao, W.X.; Yu, P.S. Heterogeneous information network embedding for recommendation. *IEEE Trans. Knowl. Data Eng.* **2019**, *31*, 357–370. [CrossRef]
46. Jamali, M.; Lakshmanan, L. HeteroMF: Recommendation in heterogeneous information networks using context dependent factor models. In Proceedings of the 22nd International Conference on World Wide Web, Rio de Janeiro, Brazil, 13–17 May 2013; Association for Computing Machinery: New York, NY, USA, 2013; pp. 643–654.
47. Suo, X.; Wei, F.; Yu, K. Entity Recommendation Via Integrating Multiple Types of Implicit Feedback in Heterogeneous Information Network. In Proceedings of the IEEE International Conference on Data Mining Workshops, New Orleans, LA, USA, 18–21 November 2017; IEEE Computer Society: Washington, DC, USA, 2017; pp. 781–786.
48. Zhu, X.; Zhang, Y.Q.; Hui, Q.Y. An Academic Literature Recommendation Method Based on Disciplinary Heterogeneous Knowledge Network. *Libr. J.* **2020**, *39*, 103–110.
49. Niu, Y.Q.; Meng, Y.Y.; Niu, Q.F. Text Recommendation Based on Heterogeneous Attention Recurrent Neural Network. *Comput. Eng.* **2020**, *46*, 52–59.
50. Zhao, J.L.; Zhao, Z.Y. Recommendation Algorithm Based on Heterogeneous Information Network Embedding and Attention Neural Network. *Comput. Sci.* **2021**, *48*, 72–79.
51. Wang, Y.G.; Mei, X.W. Fuzzy Recommendation Algorithm for Asymmetric Heterogeneous Information Network. *Comput. Eng. Appl.* **2020**, *56*, 74–79.
52. Sun, Y.; Han, J. Meta-path-based Search and Mining in Heterogeneous Information Networks. *Tsinghua Sci. Technol.* **2013**, *18*, 329–338. [CrossRef]
53. Sun, Y.; Han, J.; Yan, X.; Yu, P.S.; Wu, T. PathSim: Meta Path-based Top-K Similarity Search in Heterogeneous Information Networks. *Proc. VLDB Endow.* **2011**, *4*, 992–1003. [CrossRef]
54. Mikolov, T.; Sutskever, I.; Chen, K.; Corrado, G.; Dean, J. Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems*; MIT Press: Cambridge, MA, USA, 2013; pp. 3111–3119.

55. Xu, H.Y.; Wang, D.; Wang, F.H.; Wang, R.B. User relevance measure method combining latent Dirichlet allocation and meta-path analysis. *J. Comput. Appl.* **2019**, *39*, 3288–3292.
56. Mnih, A.; Salakhutdinov, R.R. Probabilistic matrix factorization. In Proceedings of the Twenty-First Annual Conference on Neural Information Processing Systems, Vancouver, BC, Canada, 3–6 December 2007; Curran Associates: New York, NY, USA, 2007; pp. 1257–1264.
57. Perozzi, B.; Al-Rfou, R.; Skiena, S. Deepwalk: Online learning of social representations. In Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, New York, NY, USA, 24–27 August 2014; ACM: New York, NY, USA, 2014.
58. Grover, A.; Leskovec, J. Node2vec: Scalable feature learning for networks. In Proceedings of the 22nd ACM SIGKDD International Conference, San Francisco, CA, USA, 13–17 August 2016; ACM: New York, NY, USA, 2016.