Applied Intelligence
https://doi.org/10.1007/s10489-021-02289-3

A user-knowledge dynamic pattern matching process and optimization strategy based on the expert knowledge recommendation system

Li Gao[1] · Yi Gan[2] · Zhen Yao[3] · Xianglei Zhang[3]

Accepted: 20 February 2021

Abstract

When automated pattern matching tools are used to execute user-knowledge pattern matching (UKPM) in the expert knowledge recommendation system (EKRS), user-knowledge matching is uncertain and the matching efficiency is low. To solve the above problems, the dynamic UKPM mathematical model is established and the "Entropy-Beta" method of crowdsourcing task assignment is designed to solve the model in the study. Firstly, the concept of Entropy is combined with crowdsourcing. The uncertainty of user-knowledge matching results is measured and the magnitude of the uncertainty is calculated. Secondly, based on the Beta distribution function, the accuracy of matching results is measured. The optimal matching results are selected and the matching results were sent to EKRS according to the matching probability. Thirdly, the knowledge recommendation process of UKPM is dynamically adjusted according to the matching probability. Finally, the comparison results of several algorithms showed that the Entropy-Beta algorithm could largely improve the accuracy, efficiency, dynamic regulation, and other performances of EKRS.

Keywords Expert knowledge recommendation system (EKRS) · User-knowledge pattern matching (UKPM) · Crowdsourcing task allocation · Entropy-Beta algorithm

1 Introduction

The expert knowledge recommendation system (EKRS) mentioned in this paper refers to a system that provides knowledge services for experts. Here, experts are treated as users and the knowledge and information required by experts is treated as assignment tasks. This EKRS is our platform built-in 2018 [1]. The expert knowledge set obtained from the Institutional Repository (IR) is linked to the core resource knowledge dataset (CRD) and then a new knowledge concept set, EKRS, is constructed by concept space mapping. A two-set conceptual space mapping model is established by network linking from CRD to IR [1]. Here, CRD includes various database resources, such as SD(Science Direct), CNKI(China National Knowledge Infrastructure), WOS(Web of Science), IEEE, ACM, and other full-text databases.

EKRS is an active recommendation system for knowledge discovery and knowledge services. It is a dynamic and timely system with constantly updating knowledge and can track the characteristics of scholars' interests and provide valuable references for researchers and learners.

Based on the EKRS platform, the study aims to establish the user-knowledge pattern matching model (UKPM) with the crowdsourcing task allocation method. However, in the user-knowledge pattern matching process, the uncertainty of user-knowledge set matching often occurs. To reduce the uncertainty of pattern matching in the EKRS system and improve

✉ Li Gao
dolly2005@163.com

Yi Gan
ganyi@usst.edu.cn

Zhen Yao
530512500@qq.com

Xianglei Zhang
1144182497@qq.com

[1] Library Department, University of Shanghai for Science and Technology, 516 Jungong Rd., Shanghai, China

[2] School of Mechanical Engineering, University of Shanghai for Science and Technology, 516 Jungong Rd., Shanghai, China

[3] School of Optical-Electrical and Computer Engineering, University of Shanghai for Science and Technology, 516 Jungong Rd., Shanghai, China

Published online: 13 March 2021

the matching accuracy and efficiency, the Entropy-Beta method of crowdsourcing task allocation is proposed in this paper.

The main contributions of this paper are summarized as follows. Firstly, the method combines the concept of crowdsourcing with Entropy, measures the uncertainty of user-knowledge matching results, and calculates the uncertainty. Secondly, the accuracy of matching results is measured based on the Beta distribution function to select the optimal matching results and the matching results are sent to EKRS according to the matching rate. Thirdly, the UKPM process is dynamic and the Entropy-Beta method allows dynamic adjustment according to the matching probability to ensure the effectiveness of the whole system. Finally, the efficiency and accuracy of the Entropy-Beta method are verified by the algorithm comparison.

The rest of the paper is organized as follows. Section 2 presents a brief review of related works. Section 3 introduces the user-knowledge matching model and optimization strategy based on EKRS. UKPM result evaluation and dynamic regulation method based on EKRS is presented in Section 4. In Section 5, the data were extracted from EKRS and CRD platforms for the algorithm performance analysis. Conclusions and discussion are presented in Section 6.

2 Related studies

Pattern matching is the computation process of the semantic correspondence between pattern elements with two patterns as the input. It plays an important role in data integration, data transformation, model management, e-commerce, semantic Web, and other fields [2–4]. We established the EKRS frame based on conceptual similarity and space mapping, but the detailed expansion of the platform (i.e. crowdsourcing task allocation, pattern matching, integrated decision-making model, etc.) was not explored [1]. The EKRS pattern matching process mainly refers to the user-knowledge information matching process, while in the matching process UKPM, matching uncertainty and low matching efficiency often occur, which affects the accuracy, efficiency, dynamic adjustment of the EKRS.

As a fundamental problem in data management, the method of pattern matching has received widespread attention and some automated pattern matching tools have been developed, such as COMA/COMA++, LSD, OntoBuilder, Cupid, and MatchZoo [5–7]. However, the matching results by these tools are uncertain.

With the development of Web technology, the new pattern matching method of crowdsourcing tasks to a group of distributed network users is widely adopted by researchers [8]. From the perspective of workload reduction, Dong-gyun Hong et al. [9] proposed a pattern matching method based on Web2.0, in which questions were selected from online communities, and answers were gainfully obtained from online community personnel. The crowdsourcing technology was used to improve the accuracy of pattern matching with the constructed pattern matching network [10]. Besides, in the random algorithm, based on the constraints between pattern matching networks, the accuracy of the answer was calculated [11]. Alireza et al. [12] proposed two crowdsourcing task allocation methods to improve the uncertain results obtaining with automated pattern matching tools: single correspondence correctness question in one time and multiple correspondence correctness questions in one time. They compared the accuracy and required time of the two crowdsourcing pattern matching methods and proved that the SCCQ (single correspondence correctness question) method was more accurate than the MCCQ (multiple correspondence correctness question) method, which reduced the required time for solving the problem.

The introduction of crowdsourcing allowed the calculation of the uncertainty of pattern matching. The accuracy of pattern matching should also be considered. Semantic matching plays an important role in the process of pattern matching in natural language [13, 14]. A semantic matching method based on DAML-S (DARPA Agent markup language for services) was proposed to conduct semantic matching of the input and output of services [15], but the method only used the inclusion relationship between different concepts as the standard of service capability matching and thus had a low accuracy. In a service matching method based on owl-s [16], a descriptive logic reasoning machine was used to carry out semantic matching reasoning, but the method also faced some problems such as low logical reasoning accuracy and weak service differentiation ability. An MLP method based on semantic distance was proposed to solve the problems of low accuracy of logical reasoning [17]. However, the MLP method only uses the method of information extraction domain to calculate the semantic distance between concepts and needs to traverse the entire structure of the concept tree before calculating the weight of the corresponding edge. Ontology defines the basic terms and their relationships that make up the vocabulary of the subject area, as well as the rules that combine these terms and relationships to define the extension of the vocabulary. When the size of the ontology is large, the efficiency of semantic distance calculation is low. The main problems in the above semantic matching methods are described below. Firstly, although the service capability matching method of descriptive logic reasoning is more accurate, its service differentiation ability is weak and the resolution is low. Secondly, although the semantic distance calculation method based on concept hierarchy tree is more accurate in terms of semantic similarity when the ontology scale is large, the computation load in the generation of the corresponding concept hierarchy tree and the calculation of the semantic distance between concepts is large, thus seriously affecting the efficiency of pattern matching. Therefore, a DeepMatch particle swarm optimization algorithm with a depth domain search improvement strategy was proposed to solve the sequence problem of matching results [18, 19].

EKRS is an active recommendation system for knowledge discovery and knowledge services. In the UKPM, it is necessary to determine the optimization strategy of overall pattern matching of the system from the perspective of the characteristic relationship between users and crowdsourcing tasks, the instability of users' knowledge requirements, and the dynamic adjustability of matching results. How does it recommend a matching result to the user system? A personalized recommendation system has attracted much attention because it can use users' history information to conduct in-depth analysis and mining of users' characteristics, interests, and other factors and then recommend information or services according to users' needs [20, 21]. RNN (Recurrent Neural Networks) has a good advantage in the application of the recommendation systems, especially in dynamic interest modeling and real-time recommendation. RNN dynamic recommendation system considers the dynamic factors in the recommendation system by learning the interest characteristics of dynamic changes, and realizes the real-time update of recommendation tasks [22]. RNN is good at processing serialized data. Compared with the neural network with vertical transmission, it focuses on the transmission of information on the horizontal timeline. In the transmission process, it can retain sufficient historical information and display dynamic temporal behaviors. Therefore, RNN plays a more prominent auxiliary role in the mining and analysis of users' interest preferences in the recommendation process. RNN can only determine that the recent interest has a more important influence than the previous interest, but the user's interest characteristics in the global scope cannot be obtained. The hybrid dynamic recommendation model (MN-HDRM) of multi-neural network realizes the fusion of long- and short-term interests LSTM (Long Short Term Memory) and comprehensively considers the roles of long- and short-term interests in the recommendation system [23].

To sum up, existing studies introduce the crowdsourcing method to solve the problem of pattern matching, and also explores the way to design the crowdsourcing task assignment and dynamically adjust the crowdsourcing matching results. Therefore, based on the consideration of the uncertainty of matching results obtained by automation tools, this paper proposes a crowdsourcing pattern matching optimization strategy, the Entropy-Beta method, to reduce uncertainty and increase the matching accuracy, and analyzes the performances of various algorithms with multiple data sets.

3 User-knowledge matching model and optimization strategy based on EKRS

3.1 User-knowledge dynamic pattern matching process

As shown in Fig. 1, the UKPM dynamic pattern matching process is mainly divided into five parts. In Part I, the EKRS platform firstly carries on knowledge clustering and then sends the optimal U-K matching results to the server. In Part II, the server allocates the matching results in the crowdsourcing platform to the user. In Part III, the user evaluates the accuracy of the matching results. In Part IV, the user sends the matching results to the EKRS platform in order of the matching probability; In Part V, EKRS dynamically adjusts UKPM based on the results.

The UKPM dynamic pattern matching method consists of two parts. Firstly, the optimal pattern matching of U-K is carried out with the Entropy method based on the EKRS platform. Secondly, with the Beta distribution, the accuracy of the crowdsourcing matching results allocated by the server is evaluated. Figure 2 is a schematic diagram of the UKPM dynamic pattern matching method. The method is described as follows. Firstly, the uncertainty of the result set obtained by U-K pattern matching is measured and the probability of the corresponding relation of attributes is calculated according to the matching items in the result set, (as shown in Step 1). Secondly, the attribute correspondence can be used as the matching result and the optimal matching result can be selected and sent to the server through EKRS, (as shown in Steps 2,3 and 4). Thirdly, the server assigns the matching results from the crowdsourcing platform to the user, who evaluates the accuracy of the matching results and sends the results to the EKRS platform in sequence (as shown in Steps 5 and 6). Fourthly, dynamic adjustment process,

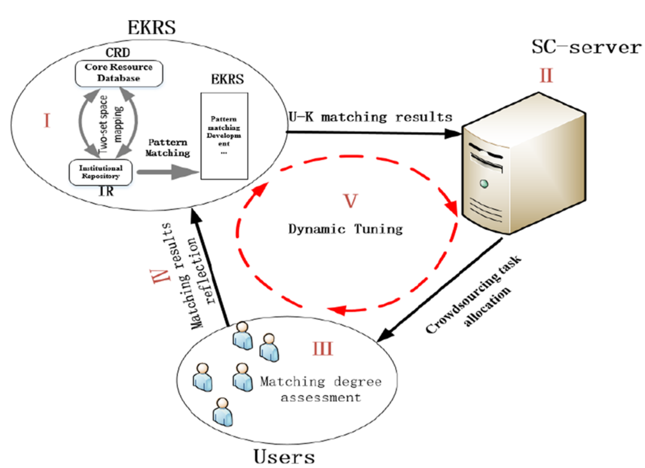
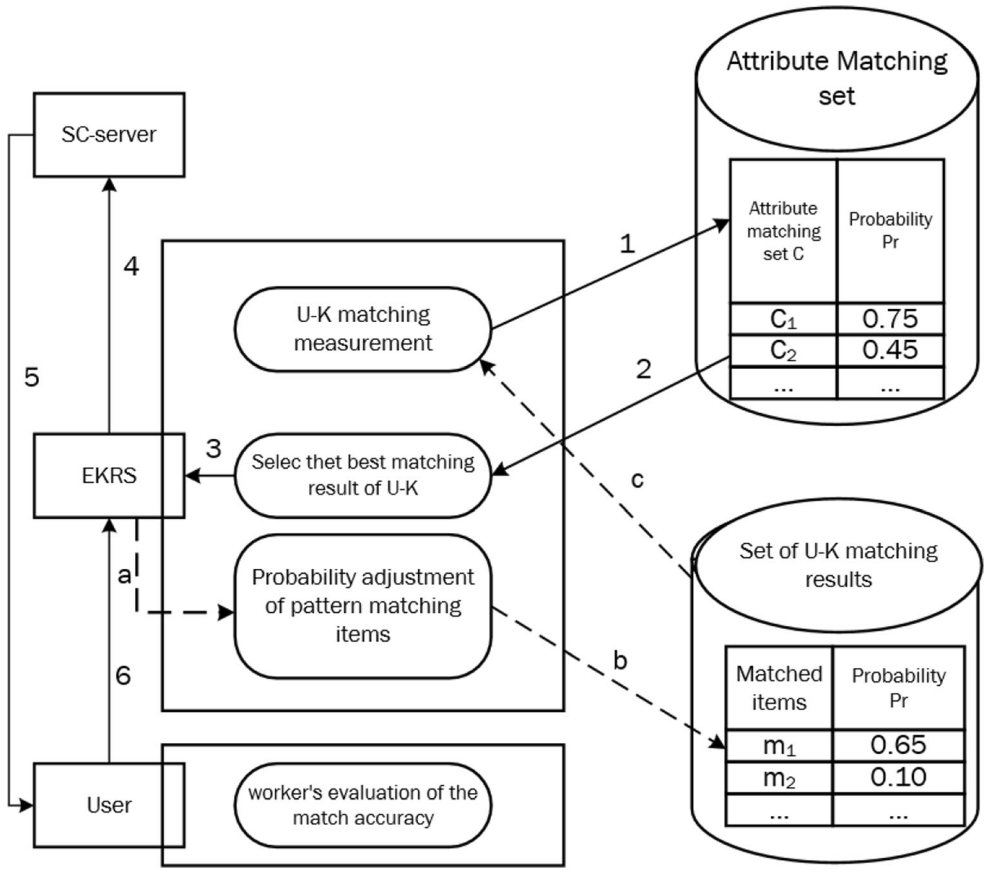


Fig. 1 UKPM dynamic pattern matching process

Fig. 2 UKPM dynamic pattern matching methods and ideas



according to the user's evaluation of matching results, the correct probability of U-K pattern matching items is adjusted, Finally, the Entropy-Beta method, is re-executed based on the adjusted result, (as shown in Dotted Lines a,b and c).

The main notations of this work are summarized in Table 1.

3.2 Mathematical model and decision variables

3.2.1 UKPM parameters and matching measurement

U denotes the user set; K denotes the knowledge set; C denotes the set of $U-K$ attribute relations, $C_i = \{(c_{u1}, c_{k1}), (c_{u2}, c_{k2})\ldots\}$,

Table 1 Notations used in the paper

Notations	Descriptions
U	User set
K	Knowledge set
C	Set of $U-K$attribute relationships
M	A user's set of $U-K$ matches
R_s	Matching result set of $U-K$
Pr	Matching rate
H	Information entropy
Q_c	Feedback on the matching accuracy of relation C of attribute $U-K$
$E(\Delta H_{Qc})$	Expectation of the result set R_s
$T(u,v)$	User's evaluation set for the matching results
A_w	Matching accuracy
f	Matching probability density function
B	U-K matching amount
b	Number of matched relationships

where (C_{u1}, C_{k1})denotes corresponding attributes;Mdenotes a set of $U-K$ matches for a user, $M_i = \{c_1, c_2, \ldots\}$, where $M_i \in C_i$. R_s denotes the $U-K$ matching result set of EKRS, $R_s = \{m_1, m_2, \ldots\}$, where$m_i \in R_s$. Pr denotes the matching rate. The correct matching result set is

$$\sum_{m_i \in R_s} P_r(m_i) = 1. \tag{1}$$

Sample: Fig. 3 shows the knowledge matching between CRD and IR database, where IR is the knowledge set representing the individual user, and CRD is the knowledge set of the repository. We select the relationship between any two (keyword, abstract, full text…") to match the relationship.

The matching results obtained after pattern matching of IR and CRD data sets $R_s = \{m_1, m_2, m_3\}$ are shown in Table 2 and $\Pr(m_1) + \Pr(m_2) + \Pr(m_3) = 1$.

In this paper, information entropy [24] is used to represent the uncertainty of the matching result set:

$$H(R_s) = -\sum_{m_i \in R_s} P_r(m_i) \mathrm{lb} P_r(m_i), \tag{2}$$

where H denotes information entropy; $P_r(m_i)$, as the random variable of the matching result, is the logarithm to the base 2 and represented by the binary system. Thus, the units of entropy are bits. The value of information entropy is the number of information entropy. The greater the uncertainty of the matching result is, the greater the entropy is. As shown in Table 1, according to Eq. (2), the uncertain result set can be calculated as:

$$H(R_s) = -0.45\mathrm{lb}0.45 - 0.30\mathrm{lb}0.30 - 0.25\mathrm{lb}0.25 = 1.54.$$

Table 2 List of pattern matching results

M_i	Pr
$m_1 = \{<c_{u1}, c_{k1}>, <c_{u2}, c_{k2}>, <c_{u3}, c_{k3}>, <c_{u4}, c_{k4}>\}$	0.45
$m_2 = \{<c_{u1}, c_{k1}>, <c_{u2}, c_{k2}>, <c_{u3}, c_{k3}>\}$	0.30
$m_3 = \{<c_{u4}, c_{k4}>, <c_{u1}, c_{k1}>, <c_{u5}, c_{k5}>\}$	0.25

3.2.2 Optimal matching result and matching probability adjustment

EKRS platform sends the matching information to the user through SC-server in a crowdsourced way. The user evaluates and analyzes the matching results and then sends the feedback information of the matching results to the EKRS platform. The feedback information should be as simple as possible. Here, the user expresses the accuracy of pattern matching results in the form of yes/no and the EKRS platform adjusts the matching probability according to the feedback (Step 5-b in Fig. 2).

Let Q_c represent the feedback on the matching accuracy of C the attribute relationship $U-K$. The matching accuracy is indicated by yes/no, so it fits the binomial Bernoulli distribution. $\Pr(Qc)$ denotes the probability of $Q_c$corresponding to yes, and $1-\Pr(Qc)$denotes the probability of $Q_c$corresponding to no. Thus, we get

$$P_A = (\Pr(Qc), 1 - \Pr(Qc)). \tag{3}$$

ΔH_{Qc} denotes the prediction of the uncertainty of the result set R_s and Eq. (2) can be rewritten as:

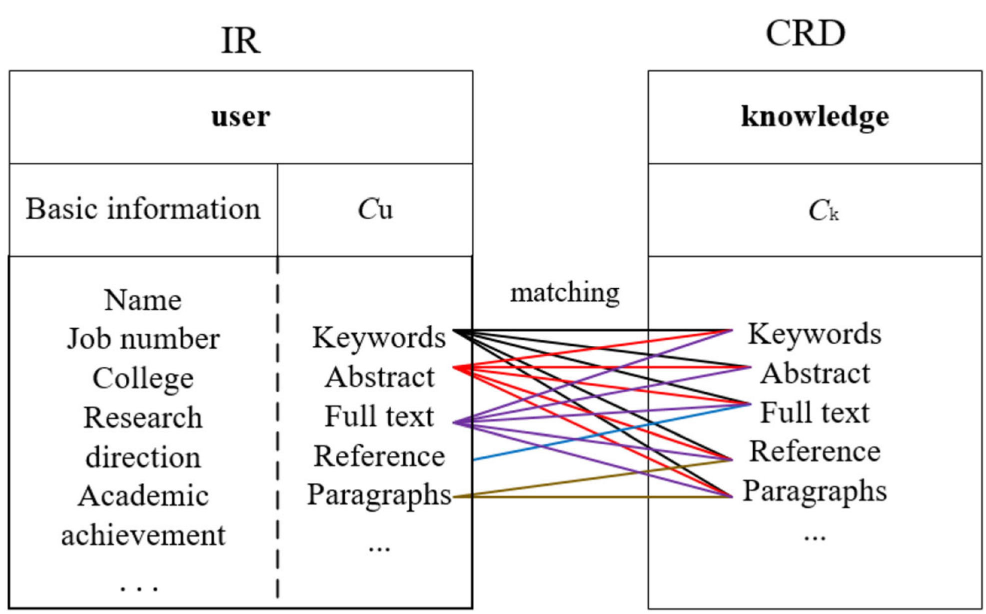


Fig. 3 Database relational matching between CRD and IR

$$\Delta H_{Qc} = H(RS) - H(RS|Qc)$$
$$= H(RS) - \left\{ -\sum_{m_i \in RS} \Pr(m_i|Qc) 1b \Pr(m_i|Qc) \right\}. \qquad (4)$$

Let $E(\Delta H_{Qc})$ represent the expectation of the result set. With information entropy, $E(\Delta H_{Qc})$ can be expressed as:

$$E(\Delta H_{Qc}) = \Pr(Qc)\left\{ H(R_S) + \sum_{m_i \in RS} \Pr(m_i|Yes) 1b \Pr(m_i|Yes) \right\} + (1 - \Pr(Qc))\left\{ H(R_S) + \sum_{m_i \in RS} \Pr(m_i|No) 1b \Pr(m_i|No) \right\}$$
$$= \Pr(Qc)\left\{ H(R_S) + \sum_{m_i \in RS} \left(\frac{\Pr(m_i)\Pr(Yes|m_i)}{\Pr(Qc)} \right) 1b \frac{\Pr(m_i)}{\Pr(Qc)} \right\} + \left\{ H(R_S) + \sum_{m_i \in RS} \left(\frac{\Pr(m_i)\Pr(No|m_i)}{1 - \Pr(Qc)} \right) 1b \frac{\Pr(m_i)}{1 - \Pr(Qc)} \right\} \times (1 - \Pr(Qc)), \qquad (5)$$

where $\Pr(Qc)$ can be obtained by superimposing the probability of the pattern matching items of the corresponding relations Qc:

$$\Pr(Qc) = \sum_{m_i \in RS \cap Qc \in m_i} \Pr(m_i). \qquad (6)$$

Sample: From the corresponding relation m_i of attributes in Table 2, the correct matching probability of the corresponding relation of attributes can be calculated. The calculation results are shown in Table 3. Taking Pr (c_2)as an example, according to Eq. (1), we get Pr (c_2)=0.45 + 0.30 = 0.75.

Suppose that $\Pr(Yes|m_i)$ and $\Pr(No|m_i)$ represent the probability that the correspondence C of the attribute under pattern match m_i is correct (wrong). The value Pr depends on whether the attribute correspondence exists in the pattern matching item m_i.

$$\Pr(m_i|Qc) = \begin{cases} 1, \mathrm{Qc} \in \mathrm{m_i} (\text{matching result conforms to the corresponding relation of the attribute}) \\ 0, Qc \notin m_i (\text{matching result doesn't conform to the corresponding relation of the attribute}) \end{cases}. \qquad (7)$$

According to Eqs. (5), (6) and (7), the expected entropy $E(\Delta H_{Qc})$ of each set of matching relation to be published can be obtained and then the matching results can be sorted according to the value of $E(\Delta H_{Qc})$. The larger $E(\Delta H_{Qc})$ indicates the higher matching degree R_s of the matching result set. Therefore, U-K matching results with a high matching degree are preferentially recommended to users. In each time, the maximum value $E(\Delta H_{Qc})$ is selected as the optimal matching result and sent to the EKRS platform. Therefore, in this paper, Eq. (6) $E(\Delta H_{Qc})$ is taken as the objective function and Eq. (7) $\Pr(m_i|\,Qc)$ is taken as the decision function.

Table 3 Attribute correspondence and correct probability

C_i	Pr
c_1=<c_{u1}, c_{k1}>	1.00
c_2=<c_{u2}, c_{k2}>	0.75
c_3=<c_{u3}, c_{k3}>	0.75
c_4=<c_{u4}, c_{k4}>	0.70
c_5=<c_{u5}, c_{k5}>	0.25

4 UKPM result evaluation and dynamic regulation method based on EKRS

4.1 Accuracy assessment of UKPM by users (part III in Fig. 1)

$T(w, v)$ denotes the set of user evaluations of the matching results, where w denotes the number of yes evaluations and v denotes the number of evaluations of no ($w > v$).

A_w denotes matching accuracy and $P_r(A_w)$ denotes the probability of the matching accuracy of the evaluation set T(u, v). As a random variable, the value A_w can be estimated according to the evaluation set $T(w, v)$. Assuming that A_w follows the Beta distribution, the probability density function $P_r(A_w)$ can be expressed as:

$$f(A_w = w) = \frac{w^{a-1}(1-w)^{b-1}}{Beta(a, b)}, \qquad (8)$$

where a and b are distribution parameters; w denotes the number of the evaluations of yes $(w > v)$. The value of $P_r(A_w)$ denotes the probability of correct matching results. Through the Bayesian analysis, $P_r(A_w)$ is obtained as follows:

$$P_r(A_w) = \frac{\Pr(<w,v>)}{\Pr(<w,v>) + \Pr(<v,w>)} = \frac{A_w{}^{u-v}}{A_w{}^{u-v} + (1-A_w)^{u-v}}. \quad (9)$$

In this paper, $E[P_r(A_w)]$ is used to represent the user's expectation of the accuracy of the matching results. Then the objective function Eq. (5) can be rewritten as:

$$E[P_r(A_w)] = \frac{\Gamma(a+w)\Gamma(b+v)}{\Gamma(a+w)\Gamma(b+v) + \Gamma(a+v)\Gamma(b+w)}. \quad (10)$$

where the Γ (n) is the Gamma function and Γ (n) = (n - 1)! holds for any positive integer n.

Proof Bayesian Theorem [25] is used to prove the above equation. $<x, y>$denotes x correct matching results and y incorrect matching results. $T(w, v)$ denotes $w$correct matches or v incorrect matches, thus $T(w, v) = <w, v> \cup <v, w>$. When $A_w = (x \sim B(\mu| a, b))$, the matching probability density function can be expressed as:

$$f(w, v|A_w = x) = \binom{w+v}{w}(x^w(1-x)^v + x^v(1-x)^w). \quad (11)$$

Then, with Bayesian Theorem, we get:

$$f_{A_w}(A_w = x|w, v) = \frac{f(w, v|A_w = x) f_{A_w}(A_w = x)}{f(w, v)} = \frac{(x^w(1-x)^v + x^v(1-x)^w)x^{a-1}(1-x)^{b-1}}{\int_0^1 (x^w(1-x)^v + x^v(1-x)^w)x^{a-1}(1-x)^{b-1}dx} = \frac{x^{a+w-1}(1-x)^{b+v-1} + x^{a+v-1}(1-x)^{b+w-1}}{\int_0^1 \left(x^{a+w-1}(1-x)^{b+v-1} + x^{a+v-1}(1-x)^{b+w-1}\right)dx}. \quad (12)$$

According to Eqs. (11) and (12), we get:

$$E[A_R] = \int_0^1 A_R f_{A_w}(A_w = x|w, v)dx = \int_0^1 \frac{x^w(1-x)^v}{x^w(1-x)^v + x^v(1-x)^w} \times \frac{\left(x^{a+w-1}(1-x)^{b+v-1} + x^{a+v-1}(1-x)^{b+w-1}\right)dx}{\int_0^1 \left(x^{a+w-1}(1-x)^{b+v-1} + x^{a+v-1}(1-x)^{b+w-1}\right)dx}$$
$$= \frac{\int_0^1 x^{a+w-1}(1-x)^{b+v-1}dx}{\int_0^1 \left(x^{a+w-1}(1-x)^{b+v-1} + x^{a+v-1}(1-x)^{b+w-1}\right)dx} = \frac{Beta(a+w, b+v)}{Beta(a+w, b+v) + Beta(a+v, b+w)} = \frac{\Gamma(a+w)\Gamma(b+v)}{\Gamma(a+w)\Gamma(b+v) + \Gamma(a+v)\Gamma(b+w)}. \quad (13)$$

According to Eq. (13), the results are only related to parameters (a and b) of Beta distribution and matching results. As for different Beta distribution parameters, as the number of matches increases, the results tend to converge and generally follow the $B(\mu| 6, 2)$ distribution, as verified in Section 5. This parameter distribution is also used in the paper.

4.2 Dynamic adjustment process (part IV in Fig. 1)

Based on the above methods, the optimal matching results are obtained and knowledge is recommended to the user. However, when the user receives the result, the user should determine the optimal matching order, the way to dynamically adjust the matching result. Given a pattern matching result set $R_s = \{m_1, m_2, \ldots\}$and problem sequence $Q_c = \{Q_1, Q_2, \ldots Q_n\}$, according to Eq. (7), it is assumed that in the matching result m_i of pattern matching item, the matching value of the problem Q_i is a, as expressed in Eq. (11). As proved above, here it is assumed that the matching value is known, $f_{A_w}(A_w = x|u, v) = a$. Thus, the correct probability of pattern matching item m_i can be adjusted by replacing $\Pr(m_i)$ with $\Pr(m_i|a)$:

$$\Pr(m_i|a) = \frac{\Pr(m_i)\Pr(a|m_i)}{\Pr(a)} = \frac{\Pr(m_i)\Pr(a)}{\Pr(Q_i)\Pr(a) + (1-\Pr(Q_i))\Pr(1-a)}. \quad (14)$$

Sample: According to Eq. (3), suppose the U-K matching result published first is c_2, then we get (c_{u2}, c_{k2}). Assuming that the matching accuracy a = 0.8, the corresponding matching error rate $1-\Pr(Qc)$=0.2, according to Eq. (14), we get:

$$\Pr(m_1|\mathrm{a}) = \frac{\Pr(m_1)\Pr(a|m_1)}{\Pr(a)} = \frac{\Pr(m_1)\Pr(0.8)}{\Pr(c_2)\Pr(0.8) + (1-\Pr(c_2))\Pr(0.2)}$$
$$= 0.45 \times \frac{0.8}{0.7 \times 0.8 + 0.3 \times 0.2} = 0.58$$

In the same way, we get $\Pr(m_2|\, a) = 0.10$ and $\Pr(m_3|\, a) = 0.32$. Thus, the entropy of the result set can be recalculated according to Eq. (2):$H(R_s) = -0.581b0.58 - 0.101b0.10 - 0.321b0.32 = 1.31$. According to the existing results, the correct probability of attribute correspondence can be obtained as: $\Pr(c1) = 0.68$, $\Pr(c2) = 0.90$, $\Pr(c3) = 1.00$, $\Pr(c4) = 0.68$, and $\Pr(c5) = 0.32$. According to Eq. (4), the order of packet distribution can be adjusted. The correct probability of matching items can be adjusted through the allocation of matching results c_2 and the evaluation of the user. According to the results, the probability that m_1 is a correct match increases, $m_1 = 0.58 > 0.45$. On the contrary, $m_2 = 0.10 < 0.30$ it indicates a reduction in the probability of a correct pattern match. The uncertainty of the entropy $H(R_s)$ of the result set is also reduced to 1.31. According to the results, the uncertainty of UKPM can be reduced by adjusting Pr and the dynamic adjustment process is effective in adjusting the optimal U-K matching.

4.3 Algorithm design

According to the above UKPM process analysis and dynamic adjustment method, the Entropy-Beta algorithm is adopted here. Let the amount of U-K match be B, and the amount of the matched relation C_i be b. When $b < \mathrm{B}$, the Entropy-Beta method is implemented until $b \geq \mathrm{B}$. According to the result of the loop, the pattern matching result with the highest probability is selected.

Algorithm 1 Entropy-Beta Algorithm

Input: Automatic pattern matching result sets $RS = \{m_1, m_2, \mathrm{K}, m_n\}$

Output: $H(RS)$, $\Pr(m_i)$, $i \in [1, n]$.

1. $b \leftarrow 1$
2. The uncertainty of the result set $H(RS)$ is calculated based on the concept of entropy.
3. Calculate the accuracy of attribute correspondence $\Pr(Qc)$.
4. Select the optimal U-K matching result.
5. The user receives the match and evaluates it.
6. The EKRS system gets feedback from all users on the UKPM results, which is $T(w, v)$.
7. The accuracy of the matching results is calculated with the Beta distribution.
8. $\forall m_i \in RS$, adjust m_i to the probability of correct pattern matching result, which is $\Pr(m_i)$.
9. If $b < B$ then
10. Go back Step 2
11. Else
12. Stop the dynamic adjustment process.
13. Return $H(RS)$ and $\Pr(m_i)$.
14. Select the pattern matching result with the highest probability m_i.

Table 4 Hardware and software configurations of the experimental platforms

Database	Hardware CPU: i5-8250U	Software Python 3.6
EKRS	Memory:8GB	Word embedding vector
	Operating System: Windows 10	
	Operating System: Windows server 2012	JDK 1.8.0_91
IR	Disk: 5 T	.NET Framework 4
		SQL Server 2016
		Tomcat 8.053
	Operating System: Windows server 2012	
CRD	Disk: SATA 19.2 TB	

Table 5 Experimental data

Database	C_{u1} keywords	C_{k1} full text	Annual total			B(Total number of matching)
			2021	2020	2019	
SD	'pattern matching'(en)	1.Secure parameterized pattern matching 2.Multiple pattern matching …… ……	11	12,472	23,484	35,867
CNKI	'pattern matching'(cn)	1.Unraveling ingredients in complex …… 2.Three-dimensional structure …… ……	0	180	713	893
EKRS	'pattern matching' (en+cn)	1.Dynamic pattern matching …… 2.Optimal pattern matching algorithms 3.A new model for pattern recognition ……	32	23,012	38,497	61,541

5 Experiments and algorithmic performance analysis

5.1 Experimental environment and data

5.1.1 Database platform hardware and software configuration

To analyze the execution efficiency of UKPM, the real data were extracted from the knowledge base application platforms of EKRS, IR, and CRD for the comparison through the simulation experiment. The operating environment of the three knowledge set platforms is shown in Table 4.

5.1.2 Pattern matching tool: MatchZoo

The automatic matching system used in this paper is MatchZoo. MatchZoo is an open-source text-matching tool developed based on TensorFlow in Python environment and can be applied in a variety of scenarios such as text retrieval, automatic question-answering, retelling questions, and dialogue system. The module provides different data generators according to different task requirements, including a single document-based data generator, a document pair-based data generator, and a document list-based data generator. Different data generators can be applied in different text matching tasks, such as text question-answering, text dialogues, and text sorting. Matchzoo was used to match various data patterns adopted in the experiment to obtain the corresponding pattern matching items and correct probability.

5.1.3 Experimental data

In the experiment, three data sets were used: the Science Direct (SD) database on the CRD platform of the university library, China National Knowledge Infrastructure (CNKI) database, and the EKRS platform constructed by mapping CRD and IR data sets. The E-Beta algorithm proposed in this paper and several common pattern matching algorithms were compared from the perspective of accuracy and efficiency.

We chose 'keywords-full text' as the highest matching relationship (C_{u1}, C_{k1}), that is, C_{u1}=' keywords',C_{k1}= 'full text'. The recommended full text and annual totals obtained by selecting 'pattern matching' are shown in Table 5(The data are derived from the statistics on the actual use of the database for the last three years).

5.2 Competitor methods

The performance analysis of UKPM algorithm mainly focuses on two processes: the process of evaluating the matching results from the user and sending feedback to EKRS (Step 5 in Fig. 2) and the performance comparison process of EKRS algorithm for the dynamic adjustment of matching results (b and c in Fig. 2, and Part V in Fig. 1).

In this paper, six popular algorithms were selected and compared with the proposed E-Beta algorithm.

As for crowdsourcing allocation, in this paper, the random algorithm [11] and the SCCQ (single correspondence correctness question) algorithm [12] were selected respectively.

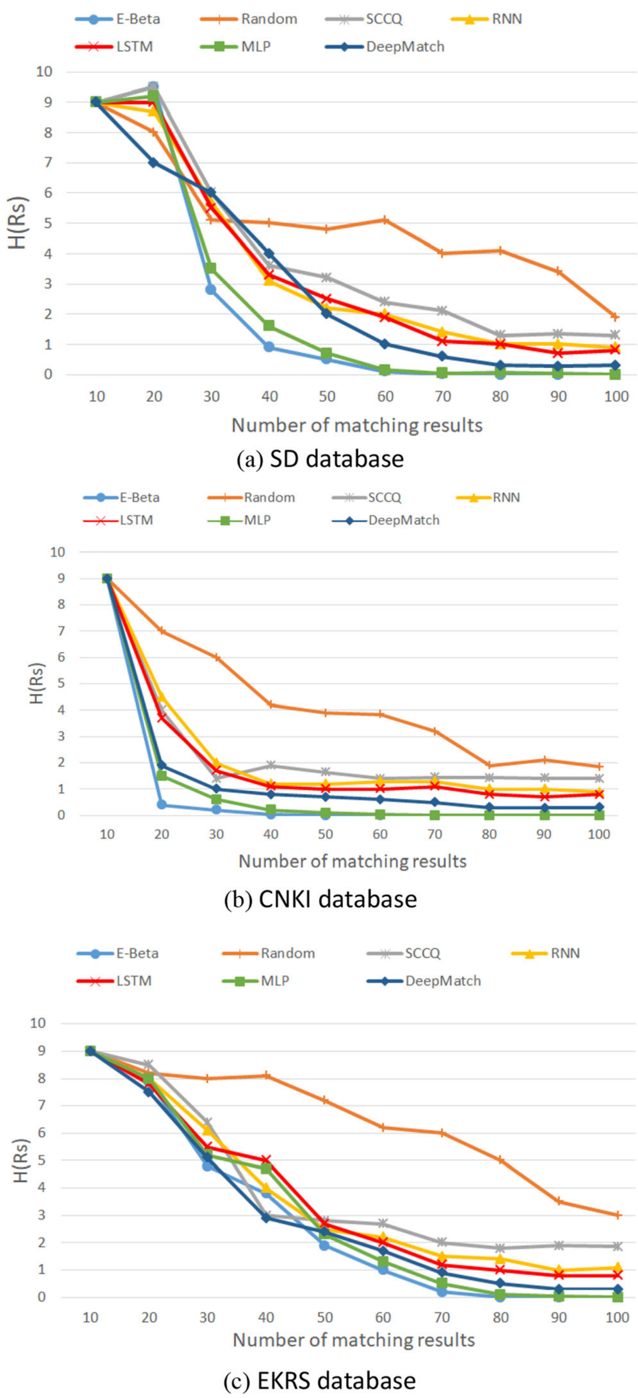


Fig. 4 Pattern matching quantity and uncertainty analysis. **a** SD database, **b** CNKI database, **c** EKRS database

Dynamic recommendation algorithm focuses on the variation of the matching process with time. In this paper, the RNN (Recurrent Neural Networks) algorithm [22] and LSTM (Long- and short-term Memory) algorithm [23] were selected respectively.

Semantic matching is important in many natural language tasks. In this paper, the multi-layer perceptron (MLP) algorithm [17] and the DeepMatch algorithm [18] were selected respectively.

All the competitor models were trained as the proposed models with the same training set and the optimal test performances of different models (e.g., the number and size of hidden layers in MLP) were compared.

5.3 Experiment I: Performance analysis of matching uncertainty and algorithm convergence

According to the E-Beta method, a set of relations (C_{u1}, C_{k1}) with the highest matching probability is first selected from the matching results $R_s = \{m_1, m_2, \ldots\}$. Second, a value is randomly assigned to the matching result $T(w, v)$ where $w + v = 100$. The Entropy-Beta method was compared with the other 6 methods to analyze the determinacy and convergence of the matching results.

Figure 4 illustrates the curve of the number of pattern matching and the uncertainty of pattern matching in 7 algorithms. Comparison results are described in the following three aspects.

Firstly, as shown in Fig. 4(a), 4(b), and 4(c), with the increase in the number of U-K pattern matches, the curve of the E-Beta method converges to 0 more early than the other 6 methods. Also, it can reduce the uncertainty of the result set of pattern matching faster and its curve is closer to 0 than the other 6 curves.

Secondly, DeepMatch and MLP were two pattern matching algorithms with the better performance. When the number of matches was small, their matching ability was strong and the convergence performance of the algorithm was good.

Thirdly, when the data set as small, the convergence and matching ability of the seven algorithms were relatively good and there was no significant difference (Fig. 4(b)). However, with the increase in data sets, the E-Beta proposed in this paper showed the better advantages (Figs. 4(a) and 4(c)).

5.4 Experiment II: Comparative accuracy analysis of algorithm matching results

According to the Entropy-Beta method, the user dynamically evaluated the accuracy of crowdsourcing matching results provided by EKRS through the server and sent the results back to the EKRS platform according to the matching rate so that it ensured that each UKPM result was the optimal choice. Figure 5 shows the matching accuracy of the seven methods. When the total amount of matching was B = 100, the seven methods were used to analyze the accuracy of the correct pattern matching results selected from the above three data sets.

As shown in Fig. 5, the curve of E-Beta is above the curves of the other two methods, indicating that the optimization

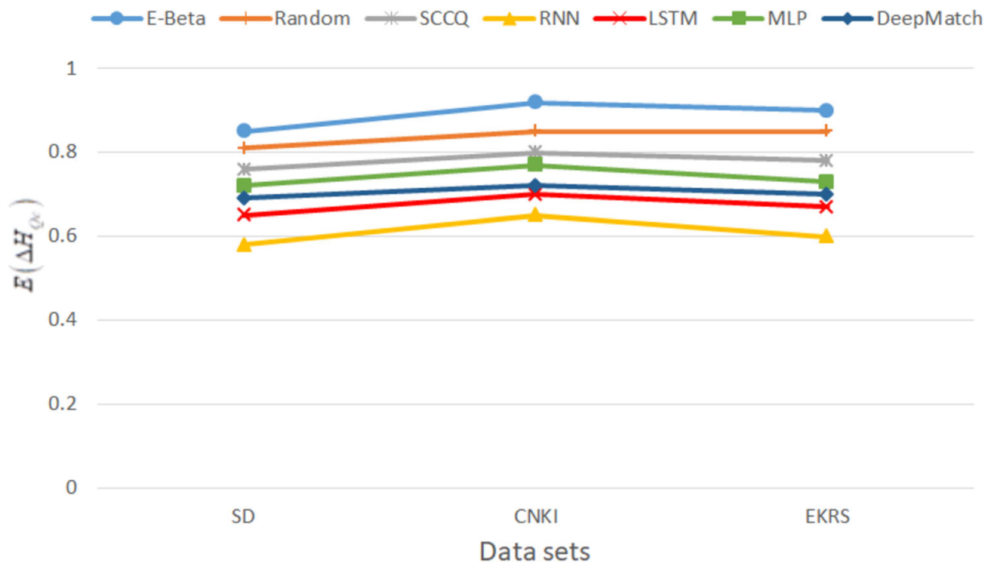


Fig. 5 Accuracy and performance analysis of crowdsourcing matching results

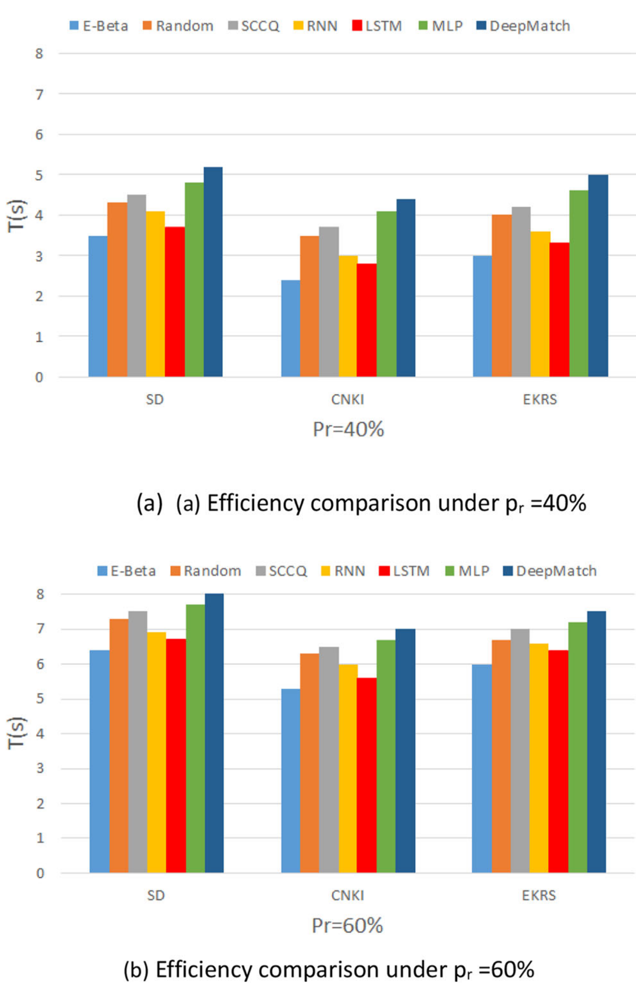


Fig. 6 Efficiency comparison of various algorithms under different matching accuracies. **a** Efficiency comparison under $p_r = 40\%$, **b** Efficiency comparison under $p_r = 60\%$

strategy of E-Beta matching results can more accurately select the correct pattern matching results. Also, the two crowdsourcing allocation algorithms (random algorithm and SCCQ algorithm), had obvious advantages in the sorting ability of U-K matching results.

5.5 Experiment III: Algorithm efficiency analysis

Based on the matching result set, the algorithm efficiency was compared under two probabilities of the correct pattern matching item (40% and 60%). The time required for the crowdsourcing matching platform to complete all the matching results is *t*. Figure 6 shows the efficiency comparison of 7 methods under different matching accuracies.

Entropy-Beta and SCCQ methods were compared with the Random method (Fig. 6). Comparison results are described in the following three aspects.

Firstly, the E-Beta method spent less time than the other 6 methods (Figs. 5(a) and 5(b), indicating that after the user sent feedback of the evaluation results to EKRS platform, the matching process was optimized with E-Beta method according to the matching accuracy.

Secondly, the two dynamic recommendation algorithms RNN and LSTM also spent less time. Also, the random algorithm and SCCQ algorithm were compared (Figs. 5 and 6). Although these two algorithms had certain advantages in the matching accuracy, they had no obvious advantage in the matching efficiency.

Thirdly, when Pr = 40%, the t of the seven algorithms was around 3 s (Fig. 6(a)). When Pr = 60%, the t of 7 algorithms was around 6 s (Fig. 6(b)). With the increase in Pr, the time of UKPM of 7 algorithms all increased. According to the comparison results between the CNKI platform with a small data set and the EKRS platform with a large data set, the time-

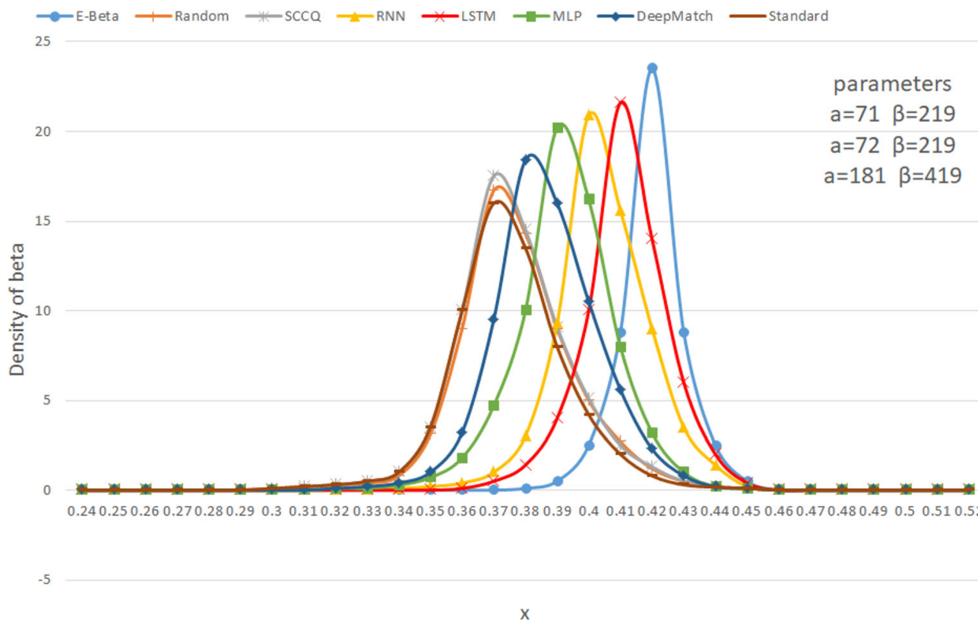


Fig. 7 Beta distribution analysis

effectiveness difference of E-Beta was less than 2.5 s, showing the better stability than other 6 methods.

5.6 Experiment IV: Dynamic adjustment process of Beta distribution analysis algorithm

The x-axis of the Beta distribution represents the value of each pattern matching rate, and the value of y corresponding to x is the probability corresponding to this matching rate. In other words, the Beta distribution can be viewed as a probability distribution of probability. It represents the probability that a matching pattern occurs. The domain of beta distribution is (0,1), which is the same as the range of the matching probability in this paper. Here we randomly assigned the match result $T(w, v)$, where w + v = 100. With the standard range from 0.30 to 0.45, we could set $\alpha = 71$, $\beta = 219$. Then we increased the matching result to 300, calculated it according to Eq. (14), and got the new Beta distribution (Fig. 7).

The dynamic adjustment process of 7 algorithms is shown in Fig. 7. And the adjustment results of the 7 methods were summarized below.

Firstly, the standard curves all became sharper and shifted to the right, indicating that the pattern match was higher than average after dynamic adjustments. Based on this new Beta distribution, we could get the mathematical expectation $E(\Delta H_{Qc})$. Therefore, when we did not know the specific value of the matching probability at the beginning, we could make some reasonable guesses that the beta distribution could be well used as a probability distribution.

Secondly, the results in Section 4.1 were only related to the parameters (a and b) of Beta distribution and matching results. Under different Beta distribution parameters, the results tended to converge with the increase in the number of matches, and generally followed the $B(\mu| 6, 2)$ distribution.

Thirdly, Compared with the curves of the other 6 methods, the curve of the E-Beta method became sharper and shifted to the right significantly, thus confirming the optimization strategy selected in the dynamic adjustment process of UKPM.

6 Conclusions and discussion

To provide experts with the latest domain knowledge according to their professional knowledge background. This paper proposes an Entropy-Beta method based on User-Knowledge Dynamic Pattern Matching Process in the EKRS platform. This method combines the concepts of crowdsourcing and Entropy to measure the uncertainty of the matching result set obtained by EKRS platform through automated tools. On this basis, the optimal matching results are selected according to the entropy of each U-K corresponding relationship relative to the result set of pattern matching, and knowledge is sent to the experts in the EKRS platform. Besides, the Beta distribution is used to calculate the accuracy of matching results and dynamically adjust the order of publishing problems according to the calculated results. SD, CNKI, and EKRS data sets in EKRS and CRD platforms were selected for the experiment. The results showed that based on the EKRS platform, the E-Beta method and optimization strategy proposed in this paper was more efficient and accurate.

At present, the automatic assignment of crowdsourcing platform tasks is still a new field. In the future, a large amount of data will be collected from the EKRS crowdsourcing platform and advanced recommendation algorithms will be applied in knowledge crowdsourcing research. The UKPM model based on EKRS crowdsourcing platform can not only recommend the professional knowledge students and experts, but also recommend optimization strategies for users according to their preferences, which can be gradually applied and popularized in business and service industries such as tourism, banking, catering and so on.

Acknowledgments The project was supported by the Ministry of Education Humanities and Social Sciences Foundation of China (20YJA870006) and the National Natural Science Foundation of China (51375314).

References

1. Gao L, Dai K, Gao LP, Jin T (2019) Expert knowledge recommendation systems based on conceptual similarity and space mapping. Expert Syst Appl 136:242–251
2. Sony KK, Rasool K (2020) Pattern matching: a quantum oriented approach. Procedia Comput Sci 167:1991–2002
3. Stylianopoulos C, Almgren M, Landsiedel O, Papatriantafilou M (2020) Multiple pattern matching for network security applications: acceleration through vectorization. J Parallel Distrib 137:34–52
4. Sukhanov S, Wu R, Debes C, Zoubir AM (2020) Dynamic pattern matching with multiple queries on large scale data streams. Inform Commun Technol 171:107–402
5. Shahmoradi S, Shouraki SB (2018) Pattern recognition tool (fuzzy elastic matching machine) and its applications in speech and handwriting recognition. Appl Soft Comput 62(1):315–327
6. Wei X, Xu L, Zhao M, Wang H (2020) Secure extended wildcard pattern matching protocol from cut-and-choose oblivious transfer. Inf Sci 529(8):132–140
7. Bayne E, Ferguson RI, Sampson AT (2018) OpenForensics: A digital forensics GPU pattern matching approach for the 21st century. Digit Investig 24(3):29–37
8. Shu J, Liu X, Zhang Y, Jiao X, Deng RH (2018) Dual-side privacy-preserving task matching for spatial crowdsourcing. J Netw Comput Appl 1231(12):101–111
9. Hong D-G, Lee Y-C, Lee J, Kim S-W (2019) CrowdStart: warming up cold-start items using crowdsourcing. Expert Syst Appl 138(12): 112813
10. Saab F, Elhajj IH, Kayssi A, Chehab A (2019) Modeling cognitive bias in crowdsourcing systems. Cogn Syst Res 58(12):1–18
11. Bhatti SS, Gao X, Chen G (2020) General framework, opportunities, and challenges for crowdsourcing techniques: a comprehensive survey. J Syst Softw 29(4):110611 online
12. Moayedikia A, Yeoh W, Ong K-L, Boo YL (2019) Improving accuracy and lowering cost in crowdsourcing through an unsupervised expertise estimation approach. Decis Support Syst 122(7): 113065
13. Fang Y, Ren Y, Park JH (2020) Semantic-enhanced discrete matrix factorization hashing for heterogeneous modal matching. Knowl-Based Syst 192(15):105381
14. Assi A, Mcheick H, Karawash A, Dhifli W (2019) Context-aware instance matching through graph embedding in lexical-semantic space. Knowl-Based Syst 186(15):104925
15. Negri E, Fumagalli L, Garetti M, Tanca L (2016) Requirements and languages for the semantic representation of manufacturing systems. Comput Ind 81(9):55–66
16. El-Ashmawi WH, Elminaam DSA, Nabil AM, Eldesouk E (2020) A chaotic owl search algorithm based bilateral negotiation model. Ain Shams Engineering Journal 7(2) Online
17. Zola P, Cortez P, Carpita M (2019) Twitter user geolocation using web country noun searches. Decis Support Syst 120(5):50–59
18. Kratzwald B, Ilić S, Kraus M, Feuerriegel S, Prendinge H (2018) Deep learning for affective computing: Text-based emotion recognition in decision support. Decis Support Syst 115(11):24–35
19. Chen L, Lu Z, Cao S, Wu Z, Cao J (2020) Personalized itinerary recommendation: Deep and collaborative learning with textual information. Expert Syst Appl 145(15):113070
20. Gupta M, Kumar P (2020) Recommendation generation using personalized weight of meta-paths in heterogeneous information networks. Eur J Oper Res 284(216):660–674
21. Lu Q, Guo F, Mall R (2019) Personalized information recommendation model based on context contribution and item correlation. Measurement 142:30–39
22. Fong IH, Li T, Fong S, Wong RK, Tallón-Ballesteros AJ (2020) Predicting concentration levels of air pollutants by transfer learning and recurrent neural network. Knowl-Based Syst 192(15):105622
23. Huo Y, Wong DF, Ni LM, Chao LS, Zhang J (2020) Knowledge modeling via contextualized representations for LSTM-based personalized exercise recommendations. Inf Sci 523:266–278
24. Wang H, Liao X, Liu Q, Zeng P (2020) Topological entropy pairs for an iterated function system. J Math Anal Appl 488(215):12476
25. Hollstein F, Prokopczuk M, Simen CW (2020) Beta uncertainty. J Bank Financ 27:105834

Li Gao the corresponding, She is doing her research work in the Library of the University of Shanghai for Science and Technology as an associate professor. Her main research interests include Algorithms and Process Modeling, Knowledge discovery and Knowledge Mining, and her Email: dolly2005@163.com