OAPRS: An Online Agriculture Prescription Recommendation System

Qingtian Zeng, Zhichao Liang, Weijian Ni, and Hua Duan

College of Information Science and Engineering,
Shandong University of Science and Technology, Qingdao 266590, China
qtzeng@163.com

Abstract. Agriculture prescription can be targeted to solve problems in agricultural production for farmers, but in reality, agricultural technology personnel (ATP) or agricultural technology experts (ATE) are also short handed. So it is difficult to ensure that all farmers' issue is resolved and the existing agricultural technology websites just tidy up some common prescriptions, mostly without involving the personalized service. So this paper designed an online agriculture prescription recommendation system which realize three functions including the automatic question answering, recommendation for ATP or ATE and recommendation for similar users based on the construction of user interest model, the problem model and prescription model. Through this system the questions put forward by farmers can be solved timely and agricultural prescriptions can be better promoted and shared, which manifests that the system has great practical value.

Keywords: agriculture prescription, user interest model, Auto QA, personalized recommendation.

1 Introduction

Agricultural technology prescription is targeted solution for problems in agricultural production. In reality, ATP or ATE's manpower is not enough, so it is difficult to ensure that all farmers' issue is resolved timely. And most existing agricultural technology websites do not involve personalized recommendation even not give farmers entrance to ask questions. Such as the basic-level agro-technique extension information platform [1] developed jointly by Ministry of Science and Technology, Ministry of Agriculture and Chinese Academy of Agricultural Sciences and Chinese gardens online system [2].

User interest model is an important part of the personalized recommendation system. For example, [3] adopted behavior sequence and their relationships to express user interest model. [4] used ontology to represent user interest model and they recorded users' behavior information especially their method distinguished between short term and long term interest model. [5] used concept network to represent user interest model. [6] put forward a fine-grained user interest modeling method which could be divided into different small user interest classes. Mao et al. [7] built user interest model with

weighted semantic network. [8] constructed user interest model based on tree vector space model.

2   Recommendation Process of the System

Fig1 shows recommendation process of the system. The system includes a total of three roles: farmer, intelligent recommendation agent, ATP or ATE.
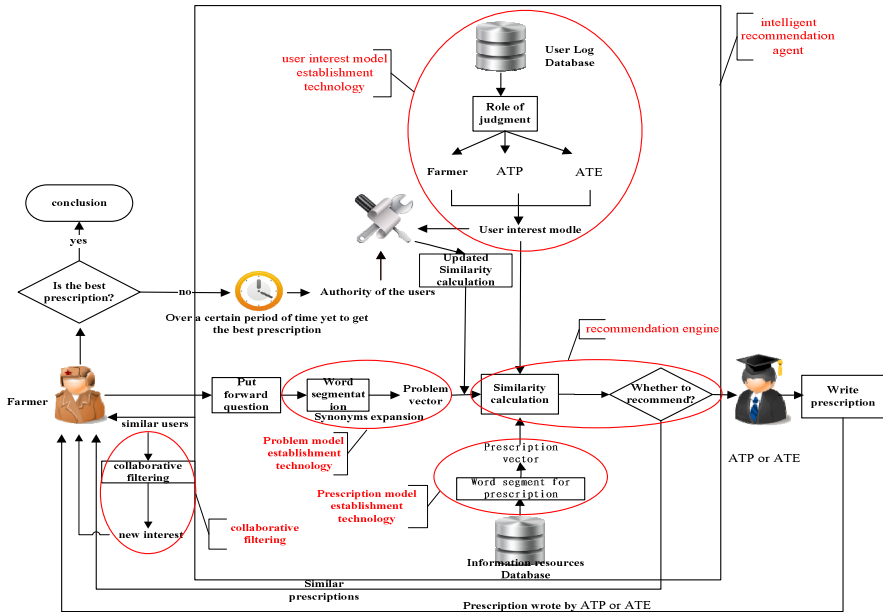

Fig. 1. Recommendation process of the system

The online agriculture prescription recommendation system has the following features:

(1) Auto QA for prescription. Prescriptions with high similarity with the question will be pushed to the farmer after he or she fills the question.

(2) Recommendation for ATP or ATE. ATP or ATE with high interest with the question will be recommended to the farmer after he or she filled the question.

(3) Recommendation for similar users. Users with high similarity with the farmer and prescriptions of mutual interest of the farmer's similar users will be recommended to the farmers when they login system and the farmers can view his or her similar users' information.

3   Key Techniques of the System

The system includes four key techniques: (1) Model construction. It comprises user interest model establishment technology, problem model establishment technology, prescription model establishment technology. (2) Word segmentation algorithm.This article adopts forward maximum matching word segmentation algorithm to segment title and text of prescriptions and questions based on domain dictionary. Word segmentation algorithm is basic of model construction. (3) Similarity calculation method. The system determines whether or not to implement recommendation according to the size of similarity. Similarity calculation method is the core of the recommendation engine in Fig1. (4) Collaborative filtering. The system uses collaborative filtering to explore new interest from similar users for the farmer.

3.1   Model Construction

(1) User Interest Model

The system automatically records the user's behavior information such as questioning behavior, browsing behavior, retrieving behavior and collecting behavior etc. and sets different weights for different behaviors.
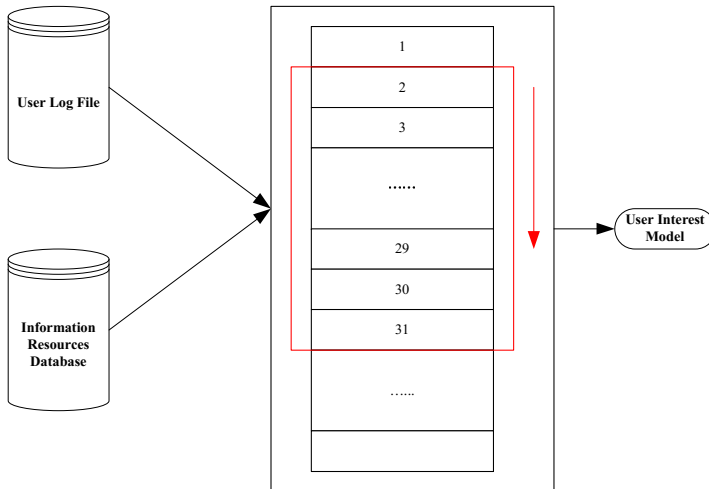

Fig. 2. Framework of data processing

Fig2 presents the framework of data processing. This article adopted window-type data processing method to mining user interest model.

The system has carried on word segmentation for the title and text of the prescription which is retrieved, browsed, collected and asked by users and then these keywords formed after word segmentation are stored into respective user log file. This article

statistically processes the data in the window and builds user interest model based on VSM (Vector Space Model). If the number of the interest node (key words) is n after statistical processing, the user interest model is expressed as following form:

$$U = \{f_1, f_2, \cdots f_n\} = \{(k_1, w_1), (k_2, w_2), \cdots, (k_n, w_n)\} \tag{1}$$

Where $f_i (1 \le i \le n)$ represents i-th feature term of user interest model. $(k_i, w_i)(1 \le i \le n)$ represents i-th feature node of user interest model. $k_i (1 \le i \le n)$ represents i-th keyword. $w_i (1 \le i \le n)$ represents the weight of the keyword k_i.

In this paper, we use two different methods to indicate the weight of keyword. Firstly word frequency. If the number of occurrences of the keyword $k_i (1 \le i \le n)$ is $count_i$, computational formula of the $w_i (1 \le i \le n)$ is as following form:

$$w_i = \frac{count_i}{n} (1 \le i \le n) \tag{2}$$

Selectly $tf*idf$. $tf*idf$ comprises two parts. The first word-based feature is the term frequency (tf).This metric measures the importance of word t_i in document, i.e. mood, d_j with $n_{i,j}$ occurrences of the word in document d_j, divided by the sum of the number of occurrences of all words in document d_j.

$$tf_{i,j} = \frac{n_{i,j}}{\sum_k n_{k,j}} \tag{3}$$

The inverse document frequency (idf) measures the importance of the word with respect to a document.

$$idf_i = \log \frac{|D|}{|\{d_j : t_i \in d_j\}|} \tag{4}$$

Then $tf*idf$ is represented as the following form:

$$tf*idf_{i,j} = tf_{i,j} * idf_i \tag{5}$$

(2) Problem Model

This article builds problem model based on Vector Space Model. The system carries on word segmentation for the title and text of the question put forward by the farmer. If the number of keyword is n after statistical processing, the problem model is expressed as following form:

$$Q = \{qf_1, qf_2, \cdots, qf_n\} = \{(qk_1, qw_1), (qk_2, qw_2), \cdots, (qk_n, qw_n)\} \tag{6}$$

Where $qf_i (1 \leq i \leq n)$ represents i-th feature term of problem model. $(qk_i, qw_i)(1 \leq i \leq n)$ represents i-th feature node of problem model. $qk_i (1 \leq i \leq n)$ represents i-th keyword. $qw_i (1 \leq i \leq n)$ represents the weight of the keyword qk_i. qw_i is calculated as the formula (2) and formula (5).

(3) Prescription Model

This article builds prescription model based on Vector Space Model. The system carries on word segmentation for the title and text of prescriptions stored in information resources database. While in the processing of word segmentation, we also perform synonym extension by using "Ci Lin" and the number of occurrences of the synonym is respectively initialized to 1. If the number of keyword is n after statistical processing, the prescription model is expressed as following form:

$$P = \{pf_1, pf_2, \cdots, pf_n\} = \{(pk_1, pw_1), (pk_2, pw_2), \cdots, (pk_n, pw_n)\} \qquad (7)$$

Where $pf_i (1 \leq i \leq n)$ represents i-th feature term of prescription model. $(pk_i, pw_i)(1 \leq i \leq n)$ represents i-th feature node of prescription model. $pk_i (1 \leq i \leq n)$ represents i-th keyword. $pw_i (1 \leq i \leq n)$ represents the weight of the keyword pk_i. pk_i is calculated as the formula (2) and formula (5).

3.2   Forward Maximum Matching Word Segmentation

Word segmentation algorithm is basis of the model construction. The domain dictionary adopted in this article is formed through training corpus in the field of agriculture and contains more than 60,000 vocabularies including 14 categories ranging from agricultural soils, agricultural meteorology, plant protection to forestry, aquaculture and so on. Algorithm 1 shows specific steps of forward maximum matching word segmentation algorithm.

Algorithm1. Forward maximum matching word segmentation algorithm
Input: *input _ str* // a string need to be carried out word segmentation.

Output: *keywords* // an array storing the keywords.

Step1: $str = string.Empty$ // a temporary variable for storing split word

   $position = 0$ //a temporary variable for storing string's location

   $k = 0$ // a temporary variable for storing matching scale's location

   $mm \_ length = 7$ // maximum word length

Step2: If $position < input \_ str.Length$, go toStep3, if not, end the algorithm and return *keywords*

Step3: If $input \_ str.Length - position >= mm \_ length$, go to Step4, if not, go to Step9.

Step4: $Length = mm \_ length - k$, if $Length == 0$, go toStep5, if not, go to Step8.

Step5: $position + +$, $k = 0$ if

$input \_ str.Length - position >= mm \_ length$, go to Step6, if not, go to Step7.

Step6: $str = input \_ str.SubString(position, mm \_ length)$.

Step7:

$str = input \_ str.SubString(position, input \_ str.Length - position)$.

Step8: $str = input \_ str.SubString(position, Length)$.

Step9: $Length = input \_ str.Length - position - k$, if $Length < 0$,

$position + +$, $k = 0$; if not, $str = input \_ str.SubString(position, Length)$.

Step10: search str in the domain dictionary, if in the dictionary then go to Step11, if not, $k + +$.

Step11: if str is the stop word, nothing to do, if not, put str into $keywords$.

Step12: if $input \_ str.Length - position >= mm \_ length$,

$position+ = mm \_ length - k$, if not,

$position+ = input \_ str.Length - position - k$.

Step13: $k = 0$, go to Step2.

3.3   Similarity Computing

In this article, we use vector cosine to measure the similarity between different models.

Based on formula (1) and formula (6), the similarity calculation formula between user interest model and question model is as follows:

$$sim(U, Q) = \frac{\sum_{i=1}^{n} w_i * qw_i}{\sqrt{\sum_{i=1}^{n} w_i^2} * \sqrt{\sum_{i=1}^{n} qw_i^2}} \tag{8}$$

If the farmer does not get a satisfactory prescription over a certain period of time, system will update the similarity calculation formula. We import authority of the ATP or ATE into the similarity calculation formula. We suppose that the number of the prescriptions filled by an ATP or ATE is w and the number of his or her prescriptions selected as the satisfactory prescription is m , then the authority calculation formula is:

$$Authority = \frac{m}{w} (0 \le m \le w) \tag{9}$$

The updated similarity calculation formula between user interest model and problem model is:

$$sim(U, Q)^{'} = \alpha Authority + (1 - \alpha)sim(U, Q) \quad (0.5 < \alpha \le 1) \tag{10}$$

Based on formula (6) and formula (7), the similarity calculation formula between question model and prescription model is as follows:

$$sim(Q,P) = \frac{\sum_{i=1}^{n} qw_i * pw_i}{\sqrt{\sum_{i=1}^{n} qw_i^2} * \sqrt{\sum_{i=1}^{n} pw_i^2}} \qquad (11)$$

Based on formula (1), the similarity calculation formula between user interest models is as follows:

$$sim(U,U') = \frac{\sum_{i=1}^{n} w_i * w_i'}{\sqrt{\sum_{i=1}^{n} w_i^2} * \sqrt{\sum_{i=1}^{n} w_i'^2}} \qquad (12)$$

3.4   Collaborative Filtering

$C = \{U_1, U_2, \cdots, U_n\}$ is the set of the users and $S = \{P_1, P_2, \cdots, P_m\}$ is the set of the prescriptions. $r_{u,p}$ represents the degree of user u's interest (similarity) in prescription P. $r_{u,p}$ is calculated through other similar users' interest in prescription P. $\widehat{U}$ is a user set with high similarity with user u. The function form of $r_{u,p}$ is as follows:

$$r_{u,p} = \frac{1}{N} \sum_{\hat{u} \in \widehat{U}} r_{\hat{u},p} \qquad (13)$$

Based on formula (1) and formula (7), the specific calculation formula of $r_{\hat{u},p}$ is as follows:

$$r_{\hat{u},p} = \frac{\sum_{i=1}^{n} w_i * pw_i}{\sqrt{\sum_{i=1}^{n} w_i^2} * \sqrt{\sum_{i=1}^{n} pw_i^2}} \qquad (14)$$

Where N represents the count of the set $\widehat{U}$. When $r_{u,p}$ is greater than the set threshold value, the system will carry on recommendation. In this way we can discover new interests for users.

3.5   Comparison between *Word Frequency* and *tf*idf*

In this article, we use two different indicators to indicate the weight of keyword. So we must compare the accuracy of two indicators. We recruit 150 system users to test the recommendation accuracy by using two different indicators. Those 150 system

users are divided into 5 groups, each group of number 10,20,30,40,50. If the users consider these prescriptions recommended by system as those they are interested in, so the recommendation is accurate. Fig3 presents the recommendation accuracy by using two different indicators and we can find that *tf*idf* has better accuracy. So in the next system implementation, we use *tf*idf* to indicate the weight of keyword.
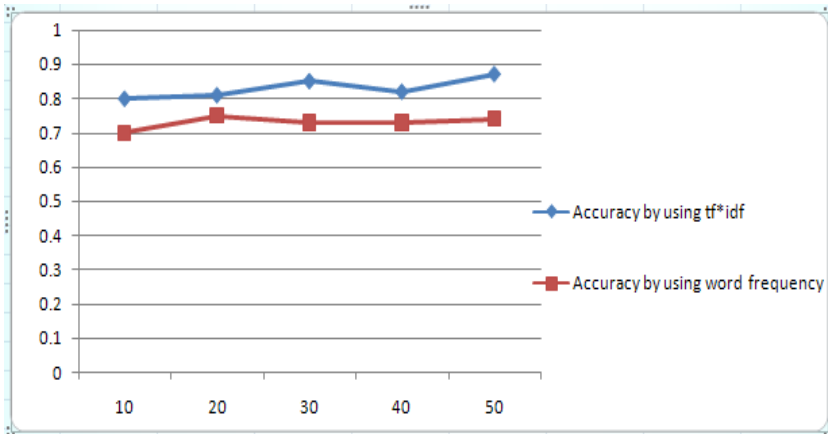

Fig. 3. Recommendation accuracy by using two different indicators

4   System Implementation

The system designed based on MVC (Model-View-Controller) model. The operating platform is Window XP or above versions and the browser is Firefox. System shown later in this article uses the data from 2013-5-1 to 2013-5-30, a total of 326 log files data. In this article, we only use recommended ATP or ATE to demonstrate our system.

4.1   View for Recommended ATP or ATE

Fig4 is the view for recommended ATP or ATE. The farmer fills title and text of the question and when he or she clicks the imagebutton 👤, system returns a histogram for showing the recommended ATP or ATE. Horizontal axis of the histogram represents the name of the ATP or ATE and vertical axis represents interest with the question of the ATP or ATE. The farmer can online communicate with the ATP or ATE.
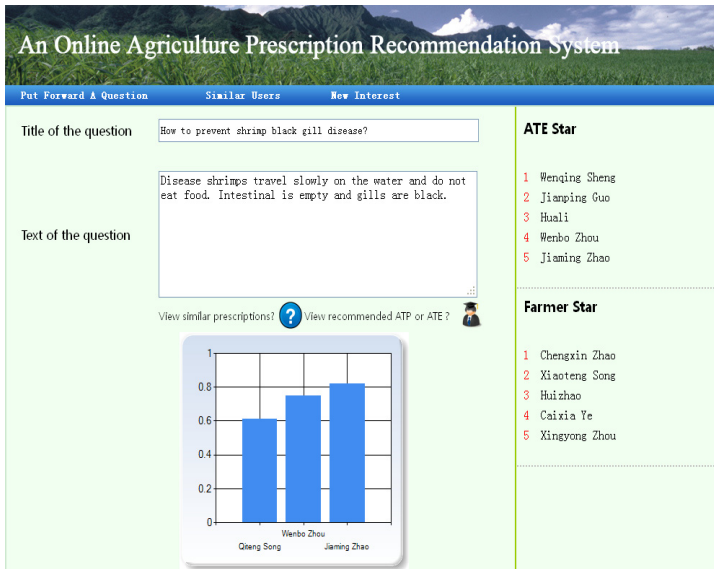
Fig. 4. Recommended ATP or ATE

5   Conclusion

This article designed and implemented an online agriculture prescription recommendation system. Farmers may find useful information in these recommended similar prescriptions to timely solve problems in the agricultural production. Farmers can view recommended ATP or ATE and online communicate with experts to get satisfactory prescriptions timely. What's more, the farmers can view similar users and find new interest from their similar users. The system provides a quick channel for farmers to seek prescriptions and provides useful agricultural information for farmers, which manifests that the system has great practical value.

We will continue to optimize the function of the system in the subsequent work such as get dynamic evolution process of the user interest model through the study of different periods of user interest model, which can predict user interest model over a period of time in the future to push timely and effective agricultural technique information for farmers.

References

1. http://www.farmers.org.cn/
2. http://zhibao.yuanlin.com/bchList.aspx
3. Iglesias, J.A., Angelov, P.P.: Creating Evolving User Behavior Profiles Automatically.10.1109/TKDE.2011 (2011)
4. Hawalah, A., Fasli, M.: A Multi-agent System Using Ontological User Profiles for Dynamic User Modelling. In: 2011 IEEE/WIC/ACM International Conference Web Intelligence and Intelligent Agent Technology (WI-IAT) (2011)
5. Kim, H.-J., Lee, S., Lee, B.-J.: Building Concept Network-Based User Profile for Personalized Web Search. 10.1109/ICIS.2010.56 (2010)
6. Tang, X., Zeng, Q.: Keyword clustering for user interest profiling refinement within paper recommender systems. Journal of Systems and Software 85(1), 87–101 (2012)
7. Mao, X., Xue, A., Ju, S.: A user interest building method based on weighted semantic net and effective information. Application Research of Computers 27(9) (2010)
8. Fei, H., Jiang, Y., Xu, L.: User interest model based on Tree Vector Space Model. Computer Technology and Development 19(5) (2009)