Modeling Leading Users in Professional Social-network Based Community Question Answering Services

Siqi SONG, Wendong WANG*, Xirong QUE, Ye TIAN, Wenwen HAN

State Key Laboratory of Networking and Switching, Beijing University of Posts and Telecommunications, Beijing 100876, China

Abstract

Differing from other traditional question-centric CQA services, professional social-network based CQA services lay a lot of emphasis on the quality of the content and social interactions within community. Besides questioning and answering, people pay more attention to authoritative and social reputation, especially some leading users. Others appreciate their contributions, and would like to make friends with them. Thus, in this study, we focus on identifying leading users within a specific topic area of professional user-centric CQA services. A leading user detecting model is proposed. Firstly, it extracts remarkable features of users’ leading ability, and then these features are exploited to train a classifier with multiple kernel learning (MKL) algorithms. Experimental results show the effectiveness of the proposed method.

Keywords: Community Question Answering; Leading Capacity; Classification; Multiple Kernel Learning

1 Introduction

Community Question Answering (CQA) service is a sophisticated Web 2.0 technology that allows users to raise questions and answer the questions asked by other people. It performs better than Web search engines in satisfying users’ personalized demands such as opinion seeking, recommendation, open-ended questions or problems solving [5].

Generally, there are two kinds of CQA services: question-centric, represented by Yahoo! Answers and Baidu Zhidao, and user-centric, whose representatives are Quora and Zhihu. Being different from question-centric CQA services, user-centric CQA websites are better at controlling the quality of the questions and answers. Besides Q&A, several new features such as real-name policy, “follow” mechanism and “PeopleRank” algorithm are designed to ensure that users acquire knowledge from primary sources and high-quality contributors in user-centric CQA services. Hence users’ intrinsic motivations for participation are enhanced because of personal satisfaction.

*Project supported by the Fundamental Research Funds for the Central Universities, the National Science and Technology Major Project under Grant (No. 2012ZX03002008) and Project on the Architecture, Key Technology Research and Demonstration of Web-based Wireless Ubiquitous Business Environment (No. 2012ZX03005008-001).

*Corresponding author.

Email addresses: songsiqi2006@126.com (Siqi SONG), wdwang@bupt.edu.cn (Wendong WANG).

1553–9105 / Copyright © 2014 Binary Information Press
DOI: 10.12733/jcis9660
April 15, 2014
in answering questions, satisfaction of curiosity, pleasure in researching new topics, and feeling competent and expert. In addition, some extra extrinsic motivations have emerged. People would like to take advantage of the social network to build reputation and social capital [11] through deliberate replies and up-votes. Along with the development of the community, some leading users who are acknowledged as authoritative and possess high social reputations stand out. As the core of a certain topic area, they make the most contributions, stimulate others’ enthusiasm to participate, and even have the ability to determine the future prosperity of the whole community.

Up to the present, a number of different approaches have been put forward in [1-5] to find proper answer providers in question-centric CQA platforms. But little work has been done for the professional social-network based CQA services. In this work, a model is proposed to automatically detect leading users in professional user-centric CQA services. Three remarkable features, namely authority, activity and influence, are selected to characterize users’ capacities to act as leading users. Then, multiple kernel learning (MKL) algorithms are employed to train a classifier, which categorizes users according to their leading abilities reflected from these aspects. There are several potential useful applications based on this approach, such as replier list recommendations for inviting to answer, subscribed resources recommendations and so on.

The key contribution of this work is that MKL algorithms are exploited to detect leading users in professional social-network based CQA services. By using multiple features as input data, the model trains a classifier to discriminate leading users from non-leading users. People used to utilize kernel machines like SVM to conduct a classification procedure. Since the features are derived from different aspects, classifier with single kernel has to keep a balance among each input feature to reach the best performance, and it may also suffer from over-fitting. The appearance of MKL algorithms alleviates this problem. Different types of data that reflect diverse remarkable features are put into different kernels, and the features can be better expressed within different kernel spaces. Experimental results on real data show that MKL methods increase the performance of the whole classifier.

The remainder of this paper is organized as follows: Section 2 details our approach, describing how leading capacity is estimated in different aspects and highlighting classification based on the evaluations from different features with MKL methods. Experimental results and discussions are presented in Section 3. At last, we conclude this paper and discuss future work in Section 4.

2 Our Approach

In this section, a leading user classification model is proposed. It mainly consists of two parts: leading capacity estimation and leading users’ classification. Making use of the input Q&A data and social data collected from a professional social-network based CQA service, users’ leading capacity is analyzed from several aspects, and then the generated data are incorporated into a multiple kernel learning (MKL) machine to train a classifier, which is ultimately used to discriminate leading users from non-leading users within a specific topic area of CQA community.

2.1 Leading capacity features

In measuring leading capacity, three pieces of information are of significance, namely authority, activity and influence. Thus, $LEADING = (ATH, ACT, INF)$. 
2.1.1 Authority

Users are supposed to be authorized, only if the following two points are met. First, they should have the ability to generate high-quality knowledge to the community. Second, their contributions should be acknowledged and approved by the crowd. Thus, \( \text{ATH} = (\text{Expertise}, \text{Satisfaction}) \).

In most cases answering to a great number of questions may probably means that one has high expertise on some topics, while asking many questions usually implies that one lacks expertise. “Z-score” [6] is such an indicator of expertise that combines one’s asking and answering patterns, as shown in Eq. (1). Though it is quite simple and only implies that an authorized user is supposed to be a content producer rather than a consumer, it is effective and practical in measuring expertise in online expertise community [6].

\[
\text{Expertise}(x) = Z\text{-score}(u) = \frac{|A(u)| - |Q(u)|}{\sqrt{|A(u)| + |Q(u)|}}
\]

where \( Q(u) \) and \( A(u) \) denote the questions that user \( u \) has asked and answered respectively.

One of the most characteristic features of CQA platforms is that users are allowed to give positive or negative judgments to others’ answers. The more supports received from the crowd, the more satisfaction the replier obtains. Therefore, the satisfaction metric of a user could be defined in Eq. (2) as the average number of votes that his replies obtained:

\[
\text{Satisfaction}(u) = \frac{|S(u)|}{|A(u)|}
\]

where \( A(u) \) denotes the questions that user \( u \) has answered, and \( S(u) \) denotes the votes that user \( u \) has received from others.

2.1.2 Activity

Activity is an important indicator to judge one’s leading capacity. Two important attributes are exploited to estimate activity, i.e. “operations” which stands for quantity, and “feedback speed” which stands for timeliness. That is, \( \text{ACT} = (\text{Operations}, \text{FeedbackSpeed}) \).

On one hand, an active user should be involved more frequently in the community than others, which results in more operations conducted within the system. Concretely, users interact with the system by raising questions, providing replies, voting to other’s resources and so on. Thus, operation could be evaluated as the linear superposition of these factors in Eq. (3).

\[
\text{Operations}(u) = |Q(u)| + |A(u)| + |V(u)|
\]

where \( Q(u) \) and \( A(u) \) denote the questions that user \( u \) has asked and answered respectively, and \( V(u) \) denotes votes that user \( u \) has given to others.

On the other hand, one’s feedback speed really counts. Although answers of high quality are preferred, people still demand a quick reply. Therefore, the time ranking of all the answers which were given to a question is introduced to calculate a user’s feedback speed, as shown in Eq. (4):

\[
\text{FeedbackSpeed}(u) = \sum_{q \in A(u)} \left[ 1 - \frac{\text{TimeRank}(q, u)}{|\text{Answers}(q)|} \right]
\]

where \( \text{TimeRank}(q, u) \) denotes the time ranking of the answer provided by user \( u \) for a question \( q \) that he replied, and \( \text{Answers}(q) \) denotes the set of answers of the question \( q \).
2.1.3 Influence

Great social influence is of great significant to evaluate one’s leading capacity. Due to the explicit “follow” mechanism, the followed users or questions can be considered as subscribed resources that may affect the followers’ behaviors. To a person, the number of followers and the number of the users who follow the questions asked or answered by him, can reflect his social influence on the mass. Therefore, \( INF = (\text{Followers}, \text{QAFollowers}) \).

In the first case, the number of followers can indicate one’s influence degree, whereupon the number of followers, denoted by \(|Fl(u)|\), is an excellent indicator of a user \( u \)’s influence.

Not restricted to follow users, the followed questions also make a difference to the followers. To a certain user \( u \), the number of the users who follow the questions asked or answered by him, denoted by \(|QAFl(u)|\), is also a measure of his influence.

2.2 Multiple kernel learning method

Support Vector Machine (SVM) is a well-known supervised machine learning method for solving binary classification problems [7]. Given a set of instance-label pairs \( \{(\vec{x}_i, y_i)\}_{i=1}^N \) where \( \vec{x}_i \) is an input vector and \( y_i \in \{-1, +1\} \) is its class label, SVM basically finds a separating hyper-plane between the two classes with the maximum margin in a higher dimensional feature space. The result of SVM learning is an \( \alpha \)-weighed linear combination of kernels with a bias \( b \):

\[
f(x) = \text{sign} \left( \sum_{i=1}^{N} \alpha_i y_i k(\vec{x}_i, \vec{x}) + b \right)
\]

where the kernel function \( k(\vec{x}_i, \vec{x}) \) computes the similarity between two examples \( \vec{x}_i \) and \( \vec{x} \). The classifier is trained by solving the following optimization problem:

\[
\begin{align*}
\text{maximize} & \quad \sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} \alpha_i \alpha_j y_i y_j k(\vec{x}_i, \vec{x}_j) \\
\text{subject to} & \quad \sum_{i=1}^{N} \alpha_i y_i = 0, \quad C \geq \alpha_i \geq 0 \quad \forall i
\end{align*}
\]

where \( C \) is a positive trade-off parameter to balance the model simplicity and classification error.

Different kernel functions are designed for different requirements and applications. Some common ones are linear kernel \( (k_{LIN}) \), polynomial kernel \( (k_{POL}) \) and Gaussian kernel \( (K_{GAU}) \). Generally, the Gaussian kernel is adopted more widely than other kernel functions because of its better performance. The key reason is that the Gaussian kernel can map the origin feature space into a quite higher, even infinite-dimensional feature space, which possesses greater flexibility and often results in a more reasonable separating hyper-plane.

\[
k_{LIN}(\vec{x}_i, \vec{x}_j) = \langle \vec{x}_i, \vec{x}_j \rangle
\]

\[
k_{POL}(\vec{x}_i, \vec{x}_j) = (\langle \vec{x}_i, \vec{x}_j \rangle)^q, \quad q \in \mathbb{N}
\]
\[ k_{GAU}(\vec{x}_i, \vec{x}_j) = \exp\left(-\frac{\|\vec{x}_i - \vec{x}_j\|^2}{2\sigma^2}\right), \quad \sigma \in \mathbb{R}_+ \quad (9) \]

In our work, the input data is derived from three features, which may be heterogeneous and non-flat distributed. Since selecting a specific kernel and its corresponding parameters may lead to bias and over-fitting, it is better to turn to the MKL algorithms. MKL exploits a set of kernel functions \( \{k_m: \mathbb{R}^{D_m} \times \mathbb{R}^{D_m} \rightarrow \mathbb{R}\}_{m=1}^M \), which take \( M \) feature representations (not necessarily different) of data instances: \( \vec{x} = \{\vec{x}^m\}_{m=1}^M \) where \( \vec{x}^m \in \mathbb{R}^{D_m} \) and \( D_m \) is the dimensionality of the corresponding feature representation. A combination function \( f_\eta: \mathbb{R}^M \rightarrow \mathbb{R} \) is proposed to combine the predefined kernels. There are different ways in which the combination function can be designed, linear, nonlinear or data-dependent. The most popular ones are the linear combination methods. In the unweighted case, we use sum or mean of the kernels as a combined one. In the weighted case, the combination function can be linearly parameterized as Eq. (10):

\[ k_\eta(\vec{x}_i, \vec{x}_j) = \sum_{m=1}^M \eta_m k_m(\vec{x}^m_i, \vec{x}^m_j) \quad (10) \]

where \( \eta_m \) donates the kernel weights. As the weights are positive, they can reveal the relative importance of each kernel. The combination parameters can also be restricted by using extra constraints, such as the \( l_p \)-norm on the kernel weights.

\[ k_\eta(\vec{x}_i, \vec{x}_j) = \left( \sum_{m=1}^M |k_m(\vec{x}^m_i, \vec{x}^m_j)|^p \right)^{1/p} \quad (11) \]

where \( p \in \mathbb{N}_+ \). Note that the MKL in the form of unweighted-sum is \( l_1 \)-norm MKL.

Synthetically utilizing each basic kernel function’s feature mapping capacity, MKL algorithms improve the performance of classification. There are two uses of MKL [8]: (a) Different kernels use the same inputs, but they correspond to different notions of similarity. Instead of trying to find which kernel works best, a supervised learning procedure does the picking job, or may use a kind of combination of them. (b) Different kernels may have inputs which come from different sources or modalities. Since different representations have different measures of similarity corresponding to different kernels, combining kernels is one possible way to combine multiple information sources. The above two options both work in our approach. For training the model based on MKL in our approach, the SHOGUN [9] toolbox is used.

3 Experiments

3.1 Experimental setup

A real-world dataset is collected from Quora from March 15th, 2012 to April 15th, 2012. In order to gather abundant data for training and testing the classifier and make the experimental results be less affected by individual subjective, the data under topic “Quora (product)” is selected. Since it is about the various features of the Quora product and what those features do and how to use them, the people who are labeled as leading users should be very familiar with these aspects and willing to devote to the community. We randomly selected 500 users that are involved in the discussions about topic “Quora (product)”. For each user, if one of the following conditions
is met, he is manually labeled as a leading user: (a) Current or former Quora staff. (b) Official site administrators, or reviewers on Quora. (c) Top writers announced by Quora. (d) Obsessive users or fans of Quora. According to these rules, 61 of the 500 users are marked as leading users. Firstly we compute the six attributes that belong to three features respectively, as described in section 2.1. All of them are normalized with the maximum value existed among all the users. Then a 10-fold cross validation procedure is employed in training and testing the performance of the MKL classification model.

Two metrics are adopted to evaluate the performance of the proposed approach: Accuracy and Area Under ROC Curve (AUC). Accuracy is a standard metric for classifiers’ performance, measuring the ratio of correctly classified examples. AUC measures the probability that a classifier would rank a randomly chosen positive instance higher than a randomly chosen negative one.

3.2 Results and analysis

Because leading user detection is regarded as a kind of two-category classification problem, the classical SVM is applied as a baseline. Then, we turn to use MKL methods, where different kernels and corresponding parameters as well as different combination strategies are utilized.

3.2.1 Effectiveness of SVM

Three common kernel functions, namely linear kernel, polynomial kernel and Gaussian kernel, are tried individually to train a SVM classifier. The corresponding parameters are refined via cross validation. As could be seen from Table 1, SVM methods achieve relatively high accuracy scores, but low AUC. It is also worthwhile to note that the Gaussian kernel turns out to perform better than the other two. This phenomenon accords with the conclusion that the Gaussian kernel often results in a more reasonable separating hyper-plane since it maps the origin feature space into a quite higher dimensional space than the other kernel functions.

<table>
<thead>
<tr>
<th>Method</th>
<th>Accuracy</th>
<th>AUC</th>
</tr>
</thead>
<tbody>
<tr>
<td>SVM (linear)</td>
<td>0.900</td>
<td>0.668</td>
</tr>
<tr>
<td>SVM (polynomial)</td>
<td>0.882</td>
<td>0.591</td>
</tr>
<tr>
<td>SVM (Gaussian)</td>
<td>0.908</td>
<td>0.696</td>
</tr>
</tbody>
</table>

3.2.2 Effectiveness of MKL

Instead of selecting a specific kernel and its corresponding parameters which have to make a trade-off to reach the best results, multiple kernels are utilized.

In the first instance, the same inputs are put into different kernels, but they represent different notions of similarity. The results are shown in Table 2. When the sum of linear kernel, polynomial kernel and Gaussian kernel is used, the performance is better than the situation that only adopts one of them individually. This is due to the integrated utilization of each kernel’s advantage enhances the interpretability of the discriminant function. Since Gaussian kernel performs better...
than the other two kernels, a unweighted combination of five Gaussian kernels with different widths \(2\sigma^2 \in \{0.001, 0.01, 0.1, 1, 10\}\) is utilized, the accuracy and AUC metrics will be upgraded again. If the weights of each Gaussian kernel are trained during cross validation, the final weight vector is set to \((0.3565, 0.0646, 0.1768, 0.3555, 0.0466)\), which implies that the Gaussian kernels with widths 0.001, 1 and 0.01 make greater contributions to the classifier. The AUC metric that reflects the capability of the classifier rises, but the accuracy drops slightly. Although the learnt weights conduct a selection procedure to highlight the more useful kernels, it does not improve the prediction, i.e. All of them are informative to our MKL models, whether there is redundant information or not [10]. At last, if we turn to use the form of \(l_p\)-norm where \(p > 1\) to combine the Gaussian kernels, the classifier performs the best. This is because they are non-sparse solutions, which account for contributions of all kernels to live up to practical applications, regardless of whether the contributions are large or small [10]. Fig. 1 illustrates the impact of the parameter \(p\). When \(p > 1\), the scores of accuracy and AUC both remain a relatively high and stable level. The accuracy metric reaches the maximum when \(p = 4\), while the AUC metric reaches the maximum when \(p = 5\). In practice, the parameter \(p\) is set to 5 due to the highest score of AUC, since AUC is more effectiveness than accuracy because it does not depend on the choice of threshold for segregating test examples into positives and negatives.

<table>
<thead>
<tr>
<th>Method</th>
<th>Accuracy</th>
<th>AUC</th>
</tr>
</thead>
<tbody>
<tr>
<td>MKL (linear+ polynomial+Gaussian, unweighted)</td>
<td>0.910</td>
<td>0.709</td>
</tr>
<tr>
<td>MKL (Gaussian, unweighted)</td>
<td>0.922</td>
<td>0.735</td>
</tr>
<tr>
<td>MKL (Gaussian, convex)</td>
<td>0.896</td>
<td>0.896</td>
</tr>
<tr>
<td>MKL (Gaussian, (l_5)-norm)</td>
<td>0.930</td>
<td>0.928</td>
</tr>
</tbody>
</table>

Since there are three distinct features to model a leading user, we put up control experiments where each feature is distributed respectively to a set of different Gaussian kernel functions. As shown in Table 3, the overall performance is also pretty good, but not as good as the former experiments, especially in the \(l_p\)-norm case. Although the three features model the leading capacity from different aspects, the data of different features are not so heterogeneous and unnormalized that they must be put into different kernels to highlight their individual effects to improve the capability of the classifier.
Table 3: Effectiveness of MKL models with inputs from different sources

<table>
<thead>
<tr>
<th>Method</th>
<th>Accuracy</th>
<th>AUC</th>
</tr>
</thead>
<tbody>
<tr>
<td>MKL (linear+ polynomial+Gaussian, unweighted)</td>
<td>0.908</td>
<td>0.709</td>
</tr>
<tr>
<td>MKL (Gaussian, unweighted)</td>
<td>0.908</td>
<td>0.763</td>
</tr>
<tr>
<td>MKL (Gaussian, convex)</td>
<td>0.904</td>
<td>0.871</td>
</tr>
<tr>
<td>MKL (Gaussian, $l_2$-norm)</td>
<td>0.914</td>
<td>0.919</td>
</tr>
</tbody>
</table>

4 Conclusions and Future Work

In this paper, we aim to discover leading users in professional user-centric CQA services. We use a leading capacity model which takes three major features into account: authority, activity and influence. Then, MKL algorithm is exploited to classify users according to their leading capacity. Experimental results show effectiveness of the proposed method in detecting leading users.

To refine the approach with the incorporation of more abundant data, several promising directions for future work exist. First, some new features will also be studied for improving the leading capacity, such as textual features and social interactions. Second, it may be attractive to distinguish the strength of users’ leading ability. We will train a multiple-category classifier based on MKL methods to categorize users into multiple leading levels.

References