Proceedings of the Twenty-Fifth AAAI Conference on Artificial Intelligence

Integrating Community Question and Answer Archives

Wei Wei[†] [*], Gao Cong[‡], Xiaoli Li[§], See-Kiong Ng[§], Guohui Li[†]
[†]School of Computer Science and Technology, Huazhong University of Science and Technology, China
[‡]School of Computer Engineering, Nanyang Technological University, Singapore
[§] Institute for Infocomm Research, Singapore
weiwei8329@gmail.com, gaocong@ntu.edu.sg, {xlli, skng}@i2r.a-star.edu.sg, guohuili@mail.hust.edu.cn

Abstract

Question and answer pairs in Community Question Answering (CQA) services are organized into hierarchical structures or taxonomies to facilitate users to find the answers for their questions conveniently. We observe that different CQA services have their own knowledge focus and used different taxonomies to organize their question and answer pairs in their archives. As there are no simple semantic mappings between the taxonomies of the CQA services, the integration of CQA services is a challenging task. The existing approaches on integrating taxonomies ignore *the hierarchical structures of the source taxonomy*. In this paper, we propose a novel approach that is capable of incorporating the parent-child and sibling information in the hierarchical structures of the source taxonomy for accurate taxonomy integration. Our experimental results with real world CQA data demonstrate that the proposed method significantly outperforms state-of-the-art methods.

Introduction

Community Question Answering (CQA) services are Internet services that enable users to ask and answer questions, as well as to search through historical question-answer pairs. Examples of such community-based knowledge sharing services include Yahoo! Answers (answers.yahoo.com), WikiAnswers (wiki.answers.com), etc. CQA services can provide an effective alterative to search engines for question answering (Xue, Jeon, and Croft 2008). For example, given a user's query question such as "*Who was the first human being on the moon?*", CQA services can return the answer "*Neil Armstrong*" directly. In contrast, a search engine will typically return a long list of ranked Web pages and the user will have to read through them to manually find the answer he/she is looking for.

The question and answer (Q&A) pairs of CQA services are typically organized into a hierarchy of categories to allow users to navigate and browse the archived questions and their answers with ease. The hierarchical categories have the following characteristics: 1) The questions in the same category or subcategory relate to the same topic. 2) The categories are arranged in a general to specific fashion where the root node contains all the categories and the leaf nodes correspond to specific categories. For example, the subcategory "Health.Dental" is a child category of "Health" in Yahoo! Answers which focuses on dental health care. 3) The hierarchies are generally organized in a way that similar categories are closer to each other (Doan et al. 2002). Such organization of the categories allows users to find what they want conveniently.

The increasing popularity of CQA services has resulted in many large-scale archives of historical Q&A pairs that can be exploited as an important knowledge resource on the Web. However, these Q&A archives cannot be easily integrated as a comprehensive Q&A archive so that users can have a better chance to find relevant answers to their questions. Instead, users often have to search (through the function of searching within a category offered by CQA services (Cao et al. 2009)) or navigate multiple unfamiliarly organized Q&A archives from different CQA services to find their answers.

We observe that different CQA services have different knowledge focus, even though their topics may overlap with each other. As such, different CQA services used different taxonomies designed to suit their particular knowledge focus to organize their Q&A pairs. As there are no simple semantic mappings between the taxonomies of different CQA services, the integration of these CQA services is not an easy task. To map the taxonomies from different CQA services using a machine learning approach, we can formulate it as a classification problem for questions: given a source taxonomy and a target taxonomy, a classifier can be built on the target taxonomy using the question documents as training data, and then classify the questions in the source taxonomy into the categories in the target taxonomy. Clearly, this method does not utilize the category information in the source taxonomy at all (questions in the same source category are likely to be mapped to similar categories in the target), and several methods (e.g., (Agrawal and Srikant 2001)) have been developed to exploit such information in the source taxonomy to improve accuracy of the mapping.

We observe that the parent-child relations and sibling relations between the categories in the source taxonomy can provide further valuable information to help characterize the question documents. However, the existing approaches do not consider them. In this paper, we propose

[*]This work was done when being a visiting student at NTU

a novel approach that is able to incorporate such implicit information (both parent-child and sibling relationships) in the source taxonomy for accurate taxonomy integration. We conducted experiments using two most popular real-world CQA archives from Yahoo! Answers and Wiki Answers. Our experimental results demonstrate that the proposed method significantly outperforms the Enhanced Naive Bayes (Agrawal and Srikant 2001).

Preliminary

We briefly describe naive Bayesian classifier (NB) for taxonomy integration. Suppose we have a set of predefined classes from the target taxonomy $\mathcal{F} = \{c_1, c_2, \ldots, c_{|\mathcal{F}|}\}$, we aim to classify a document d in $\mathcal{T}$ into a target category in $\mathcal{F}$.

Given a set of training questions D, each question is considered a set of words and each word in a document is from the vocabulary $V =< w_1, w_2, \ldots, w_{|v|} >$. To perform classification for a question d, we need to compute the posterior probability, $Pr(c_j|d), c_j \in \mathcal{F}$. Based on the Bayesian probability and the multinomial model, we have

$$Pr(c_j) = \frac{\sum_{i=1}^{|D|} Pr(c_j|d_i)}{|D|} \tag{1}$$

$$Pr(w_t|c_j) = \frac{1 + \sum_{i=1}^{|D|} N(w_t, d_i) Pr(c_j|d_i)}{|V| + \sum_{s=1}^{|V|} \sum_{i=1}^{|D|} N(w_s, d_i) Pr(c_j|d_i)} \tag{2}$$

Here, $N(w_t, d_i)$ is the number of occurrences of word w_t in question d_i. $Pr(c_j|d) \in \{0, 1\}$ depending on the class label of the question. Assuming that the probabilities of words are independent given the class, we obtain the NB classifier:

$$Pr(c_j|d) = \frac{Pr(c_j) \Pi_{k=1}^{|d|} Pr(w_{d,k}|c_j)}{\sum_{r=1}^{|C|} Pr(c_r) \Pi_{k=1}^{|d|} Pr(w_{d,k}|c_r)} \tag{3}$$

In the NB classifier, the class with the highest $Pr(c_j|d)$ is assigned as the class of d. The NB method is known to be an effective technique for text classification.

Proposed approach

We present our proposed approach that utilizes the implicit information in hierarchies of source taxonomy to integrate hierarchical CQA archives.

Table 1: Notations

Notations	Descriptions
$\mathcal{T}$	source archive
$\mathcal{F}$	target archive
v_i	a category of $\mathcal{T}$
c_j	a category of $\mathcal{F}$
d	a question of one category
$\lvert\mathcal{F}\rvert$	number of categories in $\mathcal{F}$
$\mathcal{C}(v_i)$	the child node set of v_i
$\mathcal{S}(v_i)$	the sibling node set of v_i
$\mathcal{P}(v_i)$	the parent node of v_i
$v_i^{\mathcal{T}}$	the set of nodes in the subtree rooted at v_i

Motivations

Relationships between categories CQA archives are organized by hierarchies of categories, which are usually tree structures. The state-of-the-art integration approaches make use of the data in source archive to enhance the classification performance by using the classification results of the source to adjust the classification models. The rationale behind is that the documents in the same category of source archive are similar, and thus probably should be integrated into similar categories in the target.

However, the existing approaches ignore the hierarchical structures of the source archive. The categories in a taxonomy are not independent. The hierarchical structures of a CQA archive enclose richer information than individual categories. For a category node in the source archive, we consider two types of relations in the hierarchical structure, namely parent-child and sibling. It is challenging to make use of the relationships between categories to enhance the integration since questions in two categories can be classified to either similar categories or different categories.

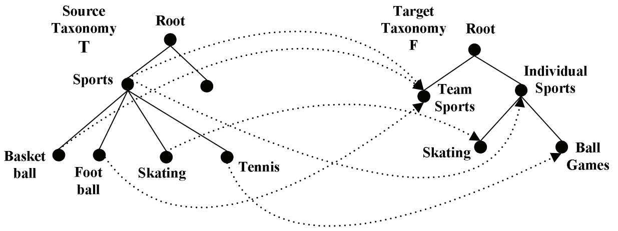

Figure 1: Category Integration Example

We illustrate this with a simplified example in Figure 1.

Consider *Sibling* relation in the source. The questions of “Football” and “Basketball” are classified into “Team Sports” category of target taxonomy. Thus we can enhance them by classifying them into the same category. On the other hand, Questions in “Tennis” or “Skating” are classified into categories “Skating” and “Ball Games” respectively, which needs a different method to improve the performance of integration by classifying them into different categories. Similarly, for parent-child relation we can observe similar phenomenon.

The relationships can be utilized in two different ways to improve the integration. First, if the questions of two categories tend to be classified into similar categories in target taxonomy, we adjust the weight of classification model to make them more likely to be classified into one category. We term this by *Similarity Re-weighting*. Second, questions of two categories are classified into different set of target categories. This motivates us to develop techniques to make use of them in a different way to reduce the probability of misclassification. We term this by *Dissimilarity Re-weighting*.

Tables 2-3 illustrate both strategies. Assume that the similarity between “Tennis” and “Basketball” is 0.01 and they need *dissimilarity re-weighting*; and the similarity between ‘Basketball” and “Football” is 0.6 and they need *similarity re-weighting* (we will elaborate how to compute similarity later on). Table 2 shows the initial prior probability of classifying a source category (corresponding to a row) into a tar-

Table 2: Initial prior probability

	Ball Games	Team Sports	Skating
Tennis	0.7	0.2	0.1
Football	0.3	0.5	0.2
Basketball	0.3	0.5	0.2

Table 3: Prior probability after adjustment

	Ball Games	Team Sports	Skating
Tennis	0.8	0.1	0.1
Football	0.24	0.66	0.1
Basketball	0.24	0.66	0.1

get category (a column). We use the probability of "Tennis" to adjust the "Football" and "Basketball" to get the adjusted prior in Table 3. With the adjusted prior, the questions of "Football" in $\mathcal{T}$ are more likely to be classified to "Team Sports" in $\mathcal{F}$, but not "Ball Games"; questions of "Tennis" in $\mathcal{T}$ are more likely to be classified to "Ball Games" in $\mathcal{F}$, but not others.

Category with mixed topics In a CQA archive, a leaf category can be represented by the questions contained in the category. However, a non-leaf category is characterized by both questions contained in the category, and questions contained in its descendant nodes. For example, a user submits a "NBA" question by navigating through the categories ("$Sports \to Basketball \to NBA$"). Although the question is more related to specific category "NBA", it is also related to generic categories "Sports" and "Basketball". Moreover, users may submit questions to upper levels although the questions are more related to lower levels since it is easier to navigate and choose the upper levels.

An overview

We extend the Bayes rule in Eq. 3 to incorporate the hierarchical information in source taxonomy $\mathcal{T}$. The posterior probability of category c_j in $\mathcal{F}$ given a question document d belonging to category v_i in $\mathcal{T}$ is computed as

$$\begin{aligned} Pr(c_j|d, \mathcal{T}^{v_i}) &= \frac{Pr(c_j)Pr(\mathcal{T}^{v_i}|c_j)Pr(d|\mathcal{T}^{v_i}, c_j)}{Pr(d, \mathcal{T}^{v_i})} \\ &\text{// the above is according to chain rule} \\ &= \frac{Pr(\mathcal{T}^{v_i})Pr(c_j|\mathcal{T}^{v_i})Pr(d|\mathcal{T}^{v_i}, c_j)}{Pr(d, \mathcal{T}^{v_i})} \\ &\propto Pr(c_j|\mathcal{T}^{v_i})Pr(d|\mathcal{T}^{v_i}, c_j) \qquad (4) \end{aligned}$$

where $\mathcal{T}^{v_i}$ represents category v_i and its hierarchical relations in $\mathcal{T}$. $Pr(\mathcal{T}^{v_i})$ and $Pr(d, \mathcal{T}^{v_i})$ are the same for all classes, and thus do not affect relative probabilities.

The remaining problem is to estimate the prior probability $Pr(c_j|\mathcal{T}^{v_i})$ and the likelihood $Pr(d|\mathcal{T}^{v_i}, c_j)$. To estimate the prior probability $Pr(c_j|\mathcal{T}^{v_i})$, we apply *Similarity Re-weighting* and *Dissimilarity Re-weighting* to make use of the relationships enclosed in a hierarchical source taxonomy. A challenge here is to identify when to use *Similarity Re-weighting* or *Dissimilarity Re-weighting*. We tackle this by developing a similarity measure. To compute similarity, we represent a leaf category with its probability distribution over the target taxonomy. For a non-leaf category that usually contains mixed topics from its descendant nodes, we take into account its descendant nodes to help to represent it. We also adjust the likelihood probability with the information of source taxonomy.

We can build a flat or a hierarchical classifier for the target CQA archive where each target category will contain the questions from source categories that are classified into it. While the proposed solution is applicable to both flat and hierarchical classifier, our experimental results show that their performance is similar. For presentation purpose, we use the flat classifier for the target archive.

Computing prior probability

We present the proposed method of incorporating the influence of child, sibling and parent nodes of a category v_i in $\mathcal{T}$ to compute its prior probability in target taxonomy. The challenge is to decide whether they play a similarity or dissimilarity re-weighting role.

The prior probability of questions in v_i being classified into category c_j in the target taxonomy $\mathcal{F}$ is computed by

$$Pr(c_j|\mathcal{T}^{v_i}) = \phi Pr(c_j) + (1 - \phi)Pr(c_j|v_i)) \qquad (5)$$

$$Pr(c_j|v_i) = \frac{\max\{0, L(c_j|v_i^{\mathcal{T}}) + L(c_j|\mathcal{C}(v_i)) + L(c_j|(\mathcal{S}(v_i) \cup \mathcal{P}(v_i)))\}}{\sum_k(\max\{0, L(c_k|v_i^{\mathcal{T}}) + L(c_k|\mathcal{C}(v_i)) + L(c_k|(\mathcal{S}(v_i) \cup \mathcal{P}(v_i)))\})} \qquad (6)$$

where $Pr(c_j)$ is the conventional prior computed by Eq.1, and $Pr(c_j|v_i)$ is the contribution from source taxonomy; $\phi(0 \leq \phi \leq 1)$ is a parameter to combine them.

Next, we discuss how to compute likelihood $L(c_j|v_i^{\mathcal{T}})$ (contribution from subtree of v_i), $L(c_j|\mathcal{C}(v_i))$ (contribution from v_i's child nodes), and $L(c_j|(\mathcal{S}(v_i) \cup \mathcal{P}(v_i)))$ (contribution from v_i's parent and sibling nodes) in Eq. 6. The contribution value can be negative (will be clear later on) and we set 0 as the minimum value.

$L(c_j|v_i^{\mathcal{T}})$**:** Recall that the descendant nodes of a non-leaf category in $\mathcal{T}$ help to characterize the category. Hence, we incorporate the probability distribution of descendant nodes into the probability distribution of a category v_i according to their similarity to v_i.

$$L(c_j|v_i^{\mathcal{T}}) = Pr(v_i \to c_j) + \sum_{v \in v_i^{\mathcal{T}} \setminus \{v_i\}} Sim(v_i, v) \times Pr(v \to c_j) \qquad (7)$$

$$p_j = Pr(v \to c_j) = \frac{|v \to c_j|}{|v|} \qquad (8)$$

where $|v \to c_j|$ is the number of questions in v that are classified to category c_j in target taxonomy.

We proceed to present the similarity function $Sim(v_i, v_j)$. A straightforward approach to calculating the similarity between two categories v_i and v_j is to represent each category by a word vector of its questions (e.g., TF-IDF) and then compute their cosine similarity.However, we aim to see if two categories are classified into similar categories in target taxonomy.

Hence, we use the probability distribution $\mathbf{p_v}$ of a category v in the target taxonomy to compute similarity. The probability distribution vector of a category v of $\mathcal{T}$ in target $\mathcal{F}$ is denoted by $\mathbf{p_v} = \{p_1, p_2, ..., p_{|\mathcal{F}|}\}$, where each p_j is calculated as Eq.8. We compute their distance by

$$Dist(v_i, v_j) = \|\mathbf{p_{v_i}} - \mathbf{p_{v_j}}\|_2 \tag{9}$$

To transform the distance measure to similarity, we use Eq.10 (Von Luxburg 2007).

$$Sim(v_i, v_j) = e^{-\frac{\|\mathbf{p_{v_i}} - \mathbf{p_{v_j}}\|^2}{2\sigma^2}} \tag{10}$$

where a larger parameter σ will result in larger similarity values, and we set $\sigma = 1.0$ (Von Luxburg 2007).

We can obtain a probability vector $\mathbf{p}_{\mathbf{v}_\mathbf{i}^{\mathcal{T}}}$ to represent $v_i^{\mathcal{T}}$ by using Eq. 7 for each c_j, $c_j \in \mathcal{F}$.

$L(c_j|\mathcal{C}(v_i))$**:** The contribution from v_i's child nodes is computed by

$$L(c_j|\mathcal{C}(v_i)) = \sum_{v_{ij} \in \mathcal{C}(v_i)} E(v_i, v_{ij}) L(c_j|(v_{ij}^{\mathcal{T}})) \tag{11}$$

Given a category v_i in $\mathcal{T}$ and its child category v_{ij}, $v_{ij} \in \mathcal{C}(v_i)$, we have

$$E(v_i, v_{ij}) = sgx(v_i, v_{ij}) \cdot Sim(v_i, v_{ij}) \tag{12}$$

$$sgx(v_i, v_{ij}) = \begin{cases} 1, & \text{if } \bar{d} > d_{ij} \text{ and } |\bar{d} - d_{ij}| > \varepsilon, \\ -1, & \text{if } \bar{d} \leq d_{ij} \text{ and } |\bar{d} - d_{ij}| > \varepsilon, \\ 0, & \text{if } |\bar{d} - d_{ij}| \leq \varepsilon, \end{cases}$$

$$\bar{d} = \frac{\|\mathbf{p_{\Omega_{v_i}}} - \mathbf{p_{v_i}}\|_2 + \|\mathbf{p_{\Omega_{v_i}}} - \mathbf{p_{v_{ij}}}\|_2}{2}$$

$$d_{ij} = \|\mathbf{p_{v_i}} - \mathbf{p_{v_{ij}}}\|_2$$

where ε is a threshold value close to 0 and is set at 0.001; $\Omega_{v_i} = \{\{v_i\} \cup \mathcal{C}(v_i)\}$=$\{v_i, v_{i1}, v_{i2}, ..., v_{i|C(v_i)|}\}\}$, and its probability distribution vector $\mathbf{p_{\Omega_c}}$ is computed by Eq.8; we set $\sigma = (\bar{d} - d_{ij})$ in computing $Sim(v_i, v_{ij})$ in Eq. 10.

Note that sign function $sgx(v_i, v_{ij})$ is used to determine whether the contribution of c_{ij} is *similarity reweighting* ($sgx(v_i, v_{ij})$=1), or *dissimilarity re-weighting* (-1), or should be ignored (0).

$L(c_j|(\mathcal{P}(v_i) \cup \mathcal{S}(v_i)))$**:** We consider parent node and sibling nodes together since their influence on v_i is similar. The influence of parent node and sibling nodes on v_i's prior probability can be computed similarly as we do for child nodes. We ignore the details due to space limitation.

$$\begin{aligned} L(c_j|(\mathcal{P}(v_i) \cup \mathcal{S}(v_i))) = {} & E(v_i, \mathcal{P}(v_i)) Pr(c_j|\mathcal{P}(v_i)) \\ & + \sum_{s_{ij} \in S(v_i)} E(v_i, s_{ij}) L(c_j|(s_{ij}^{\mathcal{T}})) \end{aligned} \tag{13}$$

We only consider the parent category $\mathcal{P}(v_i)$ of v_i, but not further ancestor categories of v_i. This is because we integrate each category in $\mathcal{T}$ in a top-down order, and thus the influence of $\mathcal{P}(v_i)$'s ancestors has already been considered when we integrate $\mathcal{P}(v_i)$. As such, the influence will be inherited when we process v_i.

Likelihood estimation

The questions of a category in source taxonomy are expected to be classified into similar categories in target taxonomy. This motivates us to use the questions of a source category to adjust the likelihood estimation as EM (Nigam et al. 2000) algorithm uses unlabeled data to modify likelihood estimation. Given a word w in source category v_i and target category c_j, we estimate $Pr(w|\mathcal{T}^{v_i}, c_j)$ by

$$Pr(w|\mathcal{T}^{v_i}, c_j) = \frac{1 + \sum_{d \in v_i} Pr^{NB}(c_j|d) N(w, d) + \sum_{d \in c_j} N(w, d)}{|V| + \sum_{s=1}^{|V|} [\sum_{d \in v_i} Pr^{NB}(c_j|d) N(w_s, d) + \sum_{d \in c_j} N(w_s, d)]} \tag{14}$$

where $Pr^{NB}(c_j|d)$ is the probability of d belonging to c_j using the NB algorithm.

CQA Integration Approach (CQai)

Given a question d in v_i of $\mathcal{T}$, the probability of d being integrated to category c_j of $\mathcal{F}$ is

$$Pr_{d \in v_i}(c_j|d) \propto Pr(c_j|\mathcal{T}^{v_i}) \times \prod [Pr(w|\mathcal{T}^{v_i}, c_j)]^{N(w,d)} \tag{15}$$

where $Pr(c_j|\mathcal{T}^{v_i})$ is computed by Eq. 5, and $Pr(w|\mathcal{T}^{v_i}, c_j)$ is computed by Eq. 14.

The algorithm, described in Algorithm 1, is called CQai.

Algorithm 1: CQA Integration (CQai)

```
Input : $\mathcal{T}$: the source taxonomy,
        $\mathcal{F}$: the target taxonomy
Result: $R$: a set of pair, $< d, c^* >, d \in \mathcal{T}, c^* \in \mathcal{F}$
$R' \leftarrow$ Use NB to classify questions in $\mathcal{T}$
// each pair of $R'$ is question and its category in $\mathcal{F}$
foreach category $v_i \in \mathcal{T}$ do
    Use Eq.7 to calculate $L(c_j|v_i^{\mathcal{T}})$ for each $c_j$ in $\mathcal{F}$
    Use Eq.11, Eq.12 and Eq.13 to calculate $L(c_j|\mathcal{C}(v_i))$,
    $L(c_j|(\mathcal{P}(v_i) \cup \mathcal{S}(v_i)))$ for each $c_j$ in $\mathcal{F}$
    Use Eq.6 and Eq.5 to calculate $Pr(c_j|\mathcal{T}^{v_i})$
    Use $R'$ and Eq.14 to update probability of each term in $\mathcal{F}$
    foreach d in $v_i$ do
        $c^* = \arg\max_{c_j} Pr(c_j|d)$ is the category of $d$;
        $Pr(c_j|d)$ is calculated with Eq.15
        add $< d, c^* >$ to $R$
Return $R$;
```

Analysis

We next analyze the rationale of the weighted function of $E(.,.)$ in the context of incorporating child node information, and the analysis equally applies to parent and sibling nodes. Recall that given a source category v_i and its child node set $\mathcal{C}(v_i)$, we construct $\Omega_c = \{v_i, v_{i1}, v_{i2}, ..., v_{i|\mathcal{C}(v_i)|}\}$.

Consider a child category v_{ij}. Let d_1 = $\|p_{\Omega_c} - p_{v_i}\|$ (the distance between v_i and Ω_c), d_2 = $\|p_{\Omega_c} - p_{v_{ij}}\|$ (the distance between c_{ij} and Ω_c), and d_3=$\|p_{v_i} - p_{v_{ij}}\|$ (the distance between v_i and v_{ij}). Recall $\bar{d} = \frac{d_1 + d_2}{2}$ in Eq. 12.

If the questions of Ω are classified into a single category in $\mathcal{F}$, it means the questions of v_i and $\mathcal{C}(v_i)$ should be classified together. Hence, $(\bar{d} - d_3)$ is close to 0. In the case, we have $\forall\ v_{ij} \in \mathcal{C}(v_i)$, $sgx(v_i, v_{ij}) = 0 \Rightarrow E(v_i,\ v_{ij})$ = 0, i.e. we disregard the influence of $\mathcal{C}(v_i)$.

We next consider that the questions of Ω are classified into several categories in $\mathcal{F}$. We have the following 4 cases.

1) $d_3 < d_1 < d_2$ or $d_3 < d_2 < d_1$: Due to $d_3 < d_1$ and $d_3 < d_2$. v_i and v_{ij} should be closer, compared with the other category in Ω. In other words, v_{ij} should play the role of *similarity enhancement* for v_i. When $d_3 < d_1 < d_2$ or $d_3 < d_2 < d_1$, we know $\bar{d} - d_3 > 0$, and thus $E(.,.)$ covers the case.

2) $d_1 < d_2 < d_3$ or $d_2 < d_1 < d_3$: Due to $d_1 < d_3$ and $d_2 < d_3$, the distribution of v_i in $\mathcal{F}$ is different from v_{ij}. Hence, v_{ij} should play the role of dissimilarity enhancement for v_i. In this case, $d_1 < d_2 < d_3$ or $d_2 < d_1 < d_3$, we have $\bar{d} - d_3 < 0$, and thus $E(.,.)$ can cover the case.

3) $d_1 < d_3 < d_2$: We rewrite $\bar{d} - d_3$ to $\frac{(d_1-d_3)+(d_2-d_3)}{2}$.

Consider the case $|d_1 - d_3| < |d_2 - d_3|$. It means v_i is more similar with v_{ij} than Ω, and thus v_{ij} should be a similarity enhancement for v_i. As $|d_1 - d_3| < |d_2 - d_3|$ and $d_1 < d_3 < d_2$, we can get $(d_3 - d_1) < (d_2 - d_3) \Rightarrow \bar{d} > d_3$, and thus $E(.,.)$ can cover the case. The above analysis is applicable to the case $|d_1 - d_3| > |d_2 - d_3|$.

4) $d_2 < d_3 < d_1$: It is similar to case 3).

Cases 1)-4) cover all the relationships between d_1, d_2, d_3. Hence, $E(.,.)$ can work well on all the cases.

Empirical Evaluation

We compare our proposed CQai algorithm with three existing state-of-the-art CQA integration methods, namely, NB, ENB and EM. We use F-score to evaluate the performance of the four integration techniques.

DataSets and Experimental Settings

We collected two real-world CQA datasets Y and W from Yahoo! Answers and Wiki Answers respectively. For Y, we used the publicly available Yahoo! Webscope Datasets, which contains 3,895,298 questions and their answers (Q&As) written in English. For W, we managed to crawl two Wiki Answers categories: Health and Sports (11/15/2010 version), and they contain 173,742 and 169,020 Q&As respectively. Additionly, Health in W has 732 subcategories, Sports has 734, and each of them has 6 levels.

Our evaluation task is to integrate the crawled Wiki Answers W into the Yahoo! Answers Y, i.e. $W \Rightarrow Y$. Also, since about a third of the questions in Wiki Answers W have answers while the rest do not (see table 4), we also test if using the answers together with the questions will enhance the integration accuracy as compared with using only the questions for the task.

Table 4: Number of Answers in Wiki Answers dataset

	Total Qs	Qs with As	Qs without As
Health	173,742	67,912	105,830
Sports	169,020	68,094	101,926

We divided Health and Sports of W into training part and test part with a ratio of $70 : 30$ (Health: 123601/50141, Sports: 118314/50706). We then randomly chose 500 questions from each test dataset and manually labeled them with the categories in Y as the ground-truth. To make the annotation manageable, we used the first two levels of taxonomy for Y, which includes 351 categories.

Note that the performance of ENB would be largely affected by its parameter ω. In our comparisons, we used the recommended ω values $(0, 1, 3, 10, 30, 100, 300, 1000)$ and report the best results for comparison. For our proposed CQai algorithm, we use the default parameter $\phi = 0.15$ in all our experiments. In fact, we found that we can always get the best results (the different Micro F-scores are less than 1%) as long as ϕ is within the range of $[0.1 - 0.3]$. With larger values of ϕ (e.g., larger than 0.5), the results become worse which indicates the importance of incorporating the hierarchical structures from source taxonomy.

Experimental Results

Table 5 and Table 6 show the comparison results among the 4 techniques on using training questions without answers and with answers, respectively.

Table 5: Experimental Results Without Answers

		NB	ENB	EM	CQai
Health	macro-F	0.3176	0.4747	0.3193	0.5793
	micro-F	0.4891	0.6142	0.4903	0.6765
Sports	macro-F	0.4318	0.4712	0.4890	0.5175
	micro-F	0.6208	0.7246	0.6831	0.7862

From Table 5, we can see that both ENB and EM performed better than NB, with improvements of $(15.72\%, 12.51\%)/(3.94\%, 10.38\%)$ and $(0.17\%, 0.12\%)/(5.72\%, 6.22\%)$ on Health/Sports datasets respectively in terms of macro-F and micro-F score. We noticed that ENB performed better than EM algorithm. One possible reason is that Wiki Answers has much less data (unlabeled) as compared with Yahoo! Answers, resulting in EM not being able to boost NB very much, especially in Health data. Our CQai algorithm significantly outperformed ENB $(10.46\%, 6.23\%)/(4.63\%, 6.16\%)$ on Health/Sports dataset in terms of macro-F and micro-F score, since our approach can leverage on additional hierarchical structural information implicit in the source taxonomy W. In addition, our method can compute the probability distribution of all possible categories in the target taxonomy Y for each category of the source taxonomy W and then exploit the relationships of the hierarchical structure to adjust the probability distribution for accurate classification.

When answers are also available, Table 6 shows that having the answers can help improve the integration performance for all the 4 techniques. This is easy to understand—the answers provide additional information about the questions. In fact, having the questions alone may not have sufficient information for accurate classification. For example, one question in the Wiki data is “Why do people crave drugs so much?” Without using its answer, NB will classify it as “Health/Other” which is wrong since the question should belong to “Health/Mental Health” category in Y. However, with the help of its answer, NB classifier is

Table 6: Experimental Results With Answers

		NB	ENB	EM	CQai
Health	macro-F	0.4980	0.7366	0.5267	0.7798
	micro-F	0.5071	0.6272	0.5256	0.7108
Sports	macro-F	0.5482	0.5900	0.6131	0.6551
	micro-F	0.6556	0.7692	0.7037	0.8038

able to make correct classification since the answer used several relevant terms about the mental health, such as "stress", "relaxed" and "pressure" etc. We also found that our CQai achieved the best results with macro-F and micro-F score $(77.98\%, 71.08\%)$ and $(65.51\%, 80.38\%)$ on Health and Sports data respectively. Note that as we are dealing with a challenging scenario which requires classifying the Q&As into 351 categories in Y, our classifier can be considered quite accurate with such performance.

Related Work

State-of-the-art taxonomy integration methods made use of the category information in the source taxonomy to improve classification results by assuming that the documents in the same category are likely to be in similar classes in the target taxonomy. Enhanced Naive Bayes (Agrawal and Srikant 2001) is based on the idea. The framework proposed by (Rajan, Punera, and Ghosh 2005), which is based on enhanced Naive Bayes, further allows the building of new categories in integration. Semi-supervised learning approaches in which the source documents are treated as additional unlabeled training data have also been employed to enhance the accuracy of the resulting classification model built on the target taxonomy. This includes the cross-training approach (Sarawagi, Chakrabarti, and Godbole 2003), bootstrapping approach (Zhang and Lee 2004a), and transductive SVM (Zhang and Lee 2004b). However, none of the above have made use of the information in the hierarchical structure of source taxonomy as we did.

Other taxonomy integration approaches have impractical assumptions for CQA integration. The GLUE system (Doan et al. 2002) performs 1-to-1 mapping from a category in the source taxonomy to a category in the target taxonomy. However, the notion of 1-to-1 mapping is too restrictive for general CQA integration. (Ichise, Takeda, and Honiden 2003) used the τ-statistic to determine whether the number of overlapping documents between two nodes from two taxonomies is high enough to consider the two nodes as identical. The method requires a significant number of common data instances between the two taxonomies as does (Zhang and Lee 2004a). Again, this requirement is impractical for integrating two CQA archives that may share few common questions. Moreover, this method also requires 1-to-1 mapping.

Other related works can be found in the area of ontology matching, in which heuristic methods are usually developed to merge the elements of the ontologies. Some of the popular ontology matching methods are the Chimaera (McGuinness et al. 2000), FCA-MERGE (Stumme and Maedche 2001), and PROMPT (Noy and Musen 2000), which require human interaction. More representative works on ontology matching can be found in (Euzenat and Shvaiko 2007).

Conclusions

Question and answer pairs in CQA archives are typically organized into categories that are hierarchically structured to facilitate users to search for the answers to their questions. Current integration approaches made use of the category information in the taxonomies for integrating different Q&A archives, but they overlook the informative parent-child and sibling relationships in the hierarchical structures of the taxonomies. This paper shows that the hierarchical structures of the categories in a source taxonomy can be exploited to enhance the integration performance. Experiments on real data showed that the proposed method CQai is more effective in integrating CQA archives than previous approaches.

Acknowledgement

We thank the anonymous reviewers for their insightful comments. Gao Cong and Wei Wei are supported in part by Microsoft Research Asia eHealth Grant and Singapore MOE AcRF Tier 1 Grant (RG16/10). Guohui Li and Wei Wei are supported in part by the National Natural Science Foundation of China (Grant No. 60873030), and the Ph.D. Program Foundation of Ministry of Education of China (Grant No. 20090142110023).

References

Agrawal, R., and Srikant, R. 2001. On integrating catalogs. In *WWW*, 603–612.

Cao, X.; Cong, G.; Cui, B.; Jensen, C. S.; and Zhang, C. 2009. The use of categorization information in language models for question retrieval. In *CIKM*, 265–274.

Doan, A.; Madhavan, J.; Domingos, P.; and Halevy, A. 2002. Learning to map between ontologies on the semantic web. In *WWW*, 662–673.

Euzenat, J., and Shvaiko, P. 2007. *Ontology matching*. Heidelberg (DE): Springer-Verlag.

Ichise, R.; Takeda, H.; and Honiden, S. 2003. Integrating multiple internet directories by instance-based learning. In *IJCAI*, 22–28.

McGuinness, D. L.; Fikes, R.; Rice, J.; and Wilder, S. 2000. The chimaera ontology environment. In *AAAI*, 1123–1124.

Nigam, K.; McCallum, A.; Thrun, S.; and Mitchell, T. 2000. Text classification from labeled and unlabeled documents using EM. *Machine learning* 39(2):103–134.

Noy, N. F., and Musen, M. A. 2000. Prompt: Algorithm and tool for automated ontology merging and alignment. In *AAAI*, 450–455.

Rajan, S.; Punera, K.; and Ghosh, J. 2005. A maximum likelihood framework for integrating taxonomies. In *AAAI*, 856–861.

Sarawagi, S.; Chakrabarti, S.; and Godbole, S. 2003. Cross-training: learning probabilistic mappings between topics. In *KDD*, 177–186.

Stumme, G., and Maedche, A. 2001. Fca-merge: Bottom-up merging of ontologies. In *IJCAI*, 225–234.

Von Luxburg, U. 2007. A tutorial on spectral clustering. *Statistics and Computing* 17(4):395–416.

Xue, X.; Jeon, J.; and Croft, W. B. 2008. Retrieval models for question and answer archives. In *SIGIR*, 475–482.

Zhang, D., and Lee, W. S. 2004a. Web taxonomy integration through co-bootstrapping. In *SIGIR*, 410 – 417.

Zhang, D., and Lee, W. S. 2004b. Web taxonomy integration using support vector machines. In *WWW*, 472 – 481.