BUPT_WILDCAT at TREC Crowdsourcing Track:

Crowdsourcing for Relevance Evaluation

Tao Xia, Chuang Zhang[1], Tai Li, Jingjing Xie
Pattern Recognition and Intelligent System Lab,
Beijing University of Posts and Telecommunications, P.R.China

terrily@hotmail.com,{zhangchuang, Lee}@bupt.edu.cn, xiejingjing113@gmail.com

Abstract

In recent years, crowdsourcing has become an effective method in many fields, such as relecance evaluation. Based on our experiment carried out in Beijing University of Posts and Telecommunications for the TREC 2011 Crowdsourcing track, in this paper we introduce our strategies in recruiting workers, obtaining their relevance and rank juegements and quality control. Then we explain the improved EM algorithm and Gaussian model that we make use of to calculate the consensus of labels. The result shows that our stategies and algorithms are effective.

1Introduction

In information retrieval, the accuracy of search engine in retrieving relevant documents is often evaluated by comparing with human judgements. The judges used to be experts, who have profound understanding in that field. However, with the ever-increasing scale of data sets to be labeled, we need a new approach to reduce the cost, time, effort and bias brought by the traditional methods, and promotes efficiency.

Recently, researches have revealed the effectiveness of crowdsourcing in dealing with the enormous data by distributing the work to a large group of "workers" or community through the Internet. In 2011 TREC crowdsourcing track, the application of crowdsourcing in search engine evaluation is addressed. The goal of this year includes:

1. How to obtain hogh-quality relevance judgements from individual crowd workers;
2. How to effectively compute consensus judgments from individual judgments;
3. Interaction between these (i.e., worker accuracy vs. subsequent consensus accuracy).[1]

Aimed at the issues above, we divide our work into two tasks. In the first part of the paper, we will explain the design and strategies of task 1(assessment). We make use of a qualification test to screen workers, and take some quality control methods to guarantee that only eligible labels are submitted. We also provide workers the opportunity to calibrate their judgements of relevance. As for the second part, we will focus on the task 2 that calculate the consensus over a set of individual

[1] Corresponding author: Chuang Zhang

worker labels. First we will review the past improvements in EM algorithm, and then introduce the algorithm we adopted, together with the detailed process and program structure. Second, we make use of the Gaussian model to estimate the workers' judgements. At the end of the notebook, we will list the result measured in a series of criteria, and compare them with the average level.

1. Asscessment

In this task, we aimed at obtaining relevance and rank judgements from workers. To guarantee the quality of labeling, we designed a qualification test on CrowdFlower to screen bad workers. Workers who passed the test are qualified to take the formal test on AMT, in which the test set data are used. All of the labels submitted by workers will be collected for further processing. Figure 1 shows the overall workflow.

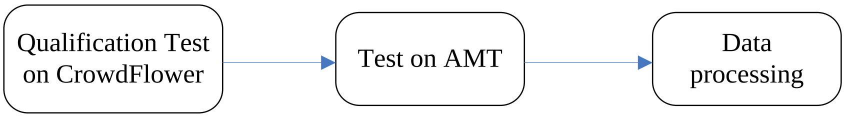


Figure 1 overall workflow

1.1 Qualification Test on CrowdFlower

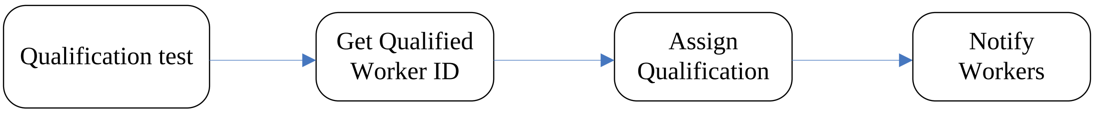


Figure 2 the workflow of qualification test

We selected 20 topic-document pairs from the judged training set. 4 of them are marked as gold (a document has reference answer provided by NIST), and their gold value are submitted to CrowdFlower. These topic-document pairs are organized into 4 jobs, each contains 5 topic-document pairs, including a gold topic-document pair.

The jobs are published on CrowdFlower, only for AMT workers. In each job, workers have two tasks to finish. Firstly, they need to judge the relevance between the topic and documents, label them as "relevant" or "irrelevant" (binary judge). And then they need to rank the 5 documents in descending order.

CrowdFlower will automatically gather the labels, compare them with the gold answers and help evaluate the ability of workers. If the worker is marked as "trust" by CrowdFlower, we will treat them as excellent workers and collect their AMT ID. Then we will invite them to take part in the HIT on AMT.

1.2 Test on AMT

In test on AMT, we scrambled the order of sets and organized the test set data into 76 HITs. Each HIT contains 6 sets. A gold set appears randomly among the 6 sets for quality control. We have set the qualification threshold to ensure only qualified workers can take the HITs. Workers finish the test on AMT as the figure shows below:

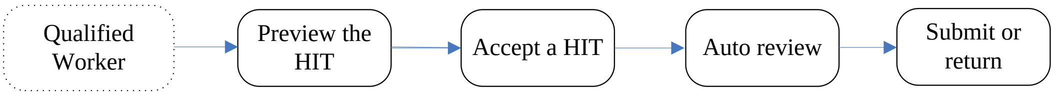


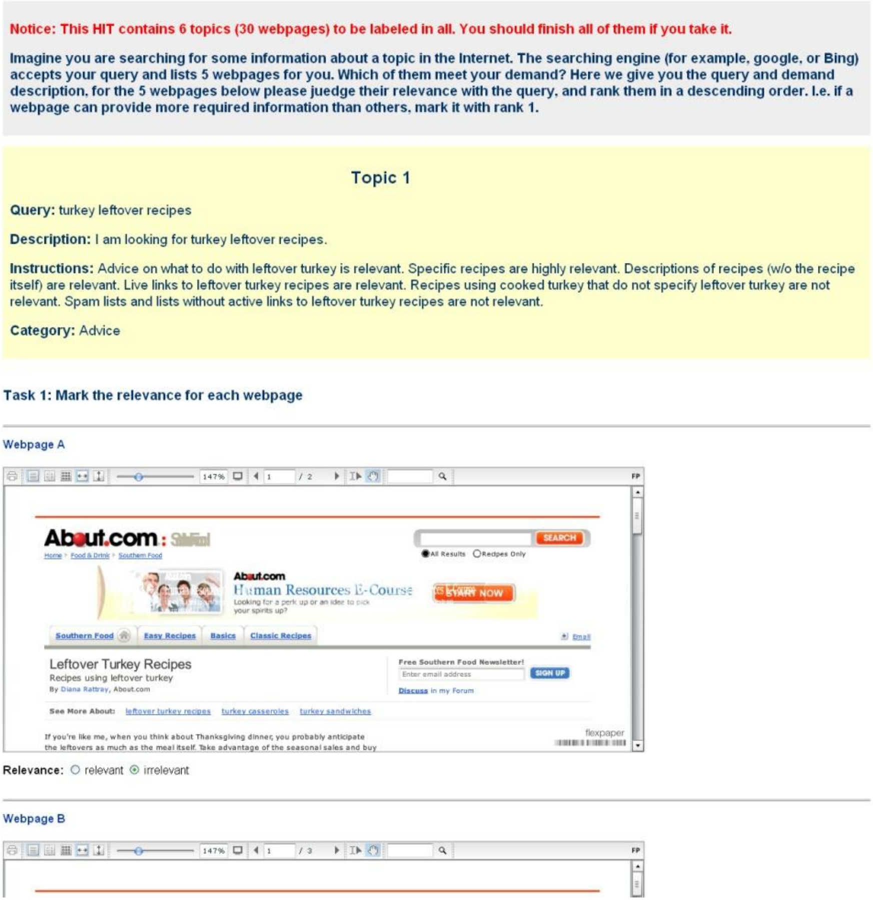
Figure 3 Workflow on AMT

1.2.1 Preview the HIT

Qualified workers are redirected to our HIT by clicking the URL provided in our email. In AMT, our HITs are shown using external webpages. This website can capture the worker ID, assignment ID and the HIT Number passed to it, shows corresponding topic-document pairs and record data. By using Flexpaper, documents are visually displayed in the webpage. Zoom and search function also provide conveniences for workers.

1.2.2 Accept a HIT

If the worker is interested in the HIT, he or she can accept the HIT and begin labeling. We have provided the query, description and instructions in the page to help them set up the judgement standard. Like the job in CrowdFlower, task 1 asks worker to judge the relevance. Mark "relevant" if the document is relevant to the query, otherwise mark "irrelevant". And task 2 is to rank the documents based on the relevance level. Among the 5 documents, the one with highest relevance ranks "1". And the others should be ranked in the descending order. Figure 4 shows the user interface of a HIT.



Notice: This HIT contains 6 topics (30 webpages) to be labeled in all. You should finish all of them if you take it.

Imagine you are searching for some information about a topic in the Internet. The searching engine (for example, google, or Bing) accepts your query and lists 5 webpages for you. Which of them meet your demand? Here we give you the query and demand description, for the 5 webpages below please juedge their relevance with the query, and rank them in a descending order. I.e. if a webpage can provide more required information than others, mark it with rank 1.

Topic 1

Query: turkey leftover recipes

Description: I am looking for turkey leftover recipes.

Instructions: Advice on what to do with leftover turkey is relevant. Specific recipes are highly relevant. Descriptions of recipes (w/o the recipe itself) are relevant. Live links to leftover turkey recipes are relevant. Recipes using cooked turkey that do not specify leftover turkey are not relevant. Spam lists and lists without active links to leftover turkey recipes are not relevant.

Category: Advice

Task 1: Mark the relevance for each webpage

Webpage A

Leftover Turkey Recipes
Recipes using leftover turkey
By Diana Rattray, About.com

See More About: leftover turkey recipes turkey casseroles turkey sandwiches

If you're like me, when you think about Thanksgiving dinner, you probably anticipate the leftovers as much as the meal itself. Take advantage of the seasonal sales and buy

Relevance: ○ relevant ◉ irrelevant

Webpage B

Figure 4 the HIT on AMT

1.2.3 Auto Review

After the workers finish all the 6 topics, they can review the wrongly labelled documents in the gold set. Document, query and other detailed information will be shown, and the corresponding correct relevance labels are highlighted. In this way, workers can get some training, and have a deeper understanding about the relevance judge standard. We believe that it will be helpful for future labeling.

1.2.4 Submit or Return

At the end of auto review page is the submit button. Sometimes it becomes a gery bar, prompting that the worker is not allowed to submit the HIT. That is because the labels provided by the worker do not meet our expection. When the worker finished the 6 topics, we will do some calculation using the labels and the time cost, and compare with our quality control threshold. If the quality meets the requirements, workers are able to submit the HIT to AMT. If it is not, the submit button will be disabled, and tell worker to return the HIT. The criteria considered will be introduced in 1.2.5. Considering some labels with very poor quality are meaningless for us, we choose to refuse them now. This stategy can effectively improve the overall quality of the judgements submitted to AMT.

1.2.5 Quality control method

We have taken some measures to control the quality of labels. When a worker is labeling, we check the compatibility of relevance and rank; when a worker finish all the topics, we calculate the binary score,NDCG score and time spent in each topic, and determine whether the worker are allowed to submit the HIT.

1) Compatibility check - Relevance and rank

The relevance and rank judgements should be valid and compatible. First, to the 5 documents of each topic, their rank value should vary from 1 to 5, and be different each other. Otherwise, a window will pop up and prompt the error. Second, the documents that are irrelevant with the query should have a lower rank value than the relevant documents. For example, document A is marked as “relevant”, while document B is marked as “irrelevant”. If the worker assign 3 to A’s rank, the rank of B should be larger than 3. If not, when “next” is pressed, a window will also tell worker about the error. Only when the relevance and rank judgements becomes compatible can the worker do the next topic.

2) Binary score

To evaluate the quality of worker’s binary judgements (relevance) dynamically, we designed a criterion – binary score. Based on the trinary label (0, 1, 2) provided by NIST, we grade workers’ binary judgements, and then normalize the sum. When worker’s judgement is correct, he or she will obtain a higher score if the judgement is wrong, a lower score is given. We hold the view that the documents with NIST answer=2 are easier to judge than the ones with NIST answer=1 or NIST answer=0. So to the docuemts NIST answer=2, if the worker mark it with 1, he or she will get fewer score; if the worker mark it with 0, he or she will get 0. Through experiments, we set the parameters as following:

- When NIST answer=0: if worker judgement=0, score +15; if worker judgement=1, score +10.
- When NIST answer=1: if worker judgement=0, score +5; if worker judgement=1, score +12.
- When NIST answer=2: if worker judgement=0, score +0; if worker judgement=1, score +9.

The sum of the scores of 5 documents in the gold set is $BV = \sum_{i=1}^{5} bv(i)$, where $bv(i)$ is the worker's score of the ith topic-document pair. Suppose all binary judgements are correct, the worker can obtain the highest score $GV = \sum_{i=1}^{5} gv(i)$, where $gv(i)$ is the worker's score of the ith topic-document pair. Hence, we define $binaryScore = \frac{BV}{GV}$.

Through the experiments in the training phase, we assign the threshold of binary score 0.85. The assignment whose binary score lower than 0.85 will be rejected to submit.

3) NDCG score

As for rank judgement, we use the NDCG score to control the quality. We refered to the NDCG algorithm in Computing Information Retrieval Performance Measures Efficiently in the Presence of Tied Scores, and altered some details to meet TREC's requirements.

Since the reference answers provided by NIST are trinary, for example, the relevance judgements by NIST of a gold set is 1, 2, 2, 0, 1, then the correct rank should be 2, 1, 1, 3, 2. If the worker's rank lables are 1, 2, 3, 4, 5 , the NDCG score is calculated as following:

- Calculate the Gain function based on the NIST answer: $G(label(v)) = 2^{label(v)} - 1$. $label(v)$ is the relevance label from NIST, i.e. 1, 2, 2, 0, 1;
- Calculate the discount function based on the worker's answer: $D_G(i) = \frac{1}{\log(1+i)}$. i is workers rank labels, i.e. 1, 2, 3, 4, 5;
- Calculate the discount function of the perfect rank label: $D_V(j) = \frac{1}{\log(1+j)}$. j is the standard rank obtained by NIST's relevance gold answer, i.e. 2, 1, 1, 3, 2;
- Calculate DCG: $DCG = \sum_{i=1}^{5} G(label(v_i)) \times D_G(i)$ .here DCG = 4.7796420679 ;
- Calculate DCV: $DCV = \sum_{j=1}^{5} G(label(v_j)) \times D_V(j)$ .here DCV = 7.2618595071429;
- Calculate NDCG: $NDCG = \frac{DCG}{DCV}$ .here NDCG = 0.6581843208681687;

Through the experiments in the training phase, we assign the threshold of NDCG score 0.62. The assignment whose NDCG score lower than 0.62 will be rejected to submit.

4) topic time

We recorded the time worker spent on each topic when the worker is labeling. For a document, we think the judgements made in less than 6 seconds are not trustworthy. As a result, the whole assignment is not allowed to submit to AMT.

2. Consensus

In this task, a set of labels contributed by individual workers were provided. We need to calculate the consensus labels from them. Both the binary judgement and rank value are required. Some of topic document pairs have NIST gold truth juegements.

2.1 EM Algorithm

Before the emerging of crowdsourcing theory, Expectation–maximization (EM) algorithm has been widely used in consensus computation. EM algorithm is an iterative approach for finding maximum likelihood estimates of parameters in statistical models, where the model depends on unobserved latent variables. In the expectation (E) step, we obtain the binary label by doing the majority decision; while in the maximization (M) step, we compute the propability of workers giving a correct label. Through sufficient iterations, the binary label approaches convergence. That is the output of EM algorithm.

Considering the workers' biases, an improved version of EM algorithm is to estimate both probabilities of givinging a correct judgement for each possible answer in the M step. This method efficiently rectifies workers' tendency to a specified label and improves the quality of output. Another version takes the difficulty level of juegements into consideration. The correct rate of workers' labels is influenced by not only the ablity of worker, but also the difficulty level of current document. So involving the two parameters into EM algorithm is a good idea.

2.1.1 Process of algorithm

The EM algorithm we used is proposed by Dawid A.P and Skene A.M. EM algorithm works as follows:

1. Given L binary labels of M topic-document pairs from N workers, for each pair D_i, initialize the correct label L_i using majority vote and save it.

2. For each worker W_j, calculate P_{cj} - the probability of labeling a pair correctly, P_{ej} - the probability of labeling incorrectly, and save them. Then set the vote weight V_j of worker W_j.

3. For each topic-document pair D_i, recalculate the correct label L_i using the vote weight of workers who have labeled the pair.

4. Repeat the step 2-3 until all the correct labels are stable.

2.1.2 Vote Weight Set

In our experiment, the worker vote weight is set as follows:

$$V_j = \log\left(\frac{P_{cj}}{P_{ej}}\right)$$

Where P_{cj} is the labeling correctly probability of worker and P_{ej} is the labeling incorrectly probability.

The vote weight comes from the following model:

For each topic-document pair D$_i$, the probability of correctly labeling L$_i$=1 is:

$$P_{i1} = \prod_{L_{ij}=1} P_{cj} \cdot \prod_{L_{ij}=0} P_{ej}$$

And P$_{i0}$ can be obtained in the same way. Then:

$$\begin{aligned}\log\left(\frac{P_{i1}}{P_{i0}}\right) &= \log\left(\frac{\prod_{L_{ij}=1} P_{cj} \cdot \prod_{L_{ij}=0} P_{ej}}{\prod_{L_{ij}=1} P_{ej} \cdot \prod_{L_{ij}=0} P_{cj}}\right) \\ &= \log\left(\prod_{L_{ij}=1} \frac{P_{cj}}{P_{ej}} \cdot \prod_{L_{ij}=0} \frac{P_{ej}}{P_{cj}}\right) = \log\left(\prod_{L_{ij}=1} \frac{P_{cj}}{P_{ej}} \cdot \prod_{L_{ij}=0} \frac{P_{ej}}{P_{cj}}\right) \\ &= \sum_{L_{ij}=1} \log\left(\frac{P_{cj}}{P_{ej}}\right) - \sum_{L_{ij}=0} \log\left(\frac{P_{cj}}{P_{ej}}\right)\end{aligned}$$

If the result of the expression above is larger than 0, i.e. P$_{i1}$>P$_{i0}$, the correct label L$_i$ tend to be 1. The equation also expresses the vote result with weight V$_j$.

2.1.3 Program

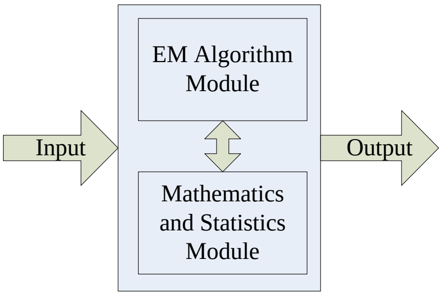


Figure 5 program of EM algorithm

Its modules include:

1. EM algorithm module: the implementation of the EM algorithm above.

2. Mathematics and Statistics module: based on the EM algorithm, provide mathematics and statistics functions.

3. Data module: read the input data and write program result into the output file.

2.2 EM Algorithm with Gaussian Model

2.2.1 Gaussian Model

There are M topic-document pairs, annotated with L binary labels by N workers. For each pair D$_i$, it has a relevant degree B$_i$ (0<B$_i$<1), and a relevant binary label L$_i$ (L$_i$=1 when B$_i$>0.5, or L$_i$=0). Each worker W$_j$ has an ability level 1/A$_j$ (A$_j$>0, its inverse is proportional to work quality of W$_j$), and a relevant label threshold T$_j$ (0<T$_j$<1, in this experiment, T$_j$=0.5).

In this model, when W$_j$ is in the annotation, he will obtain a relevant value B$_{ij}$ first. B$_{ij}$ is consistent with Gaussian distribution whose expectation is B$_i$ and variance is A$_j$. Second, W$_j$ will compare B$_{ij}$ and his relevant label threshold T$_j$. If B$_{ij}$>T$_j$, W$_j$ will label L$_{ij}$=1 to D$_i$. Otherwise W$_j$ will label L$_{ij}$ =0.

When Li=1, the probability of Lij=Li is:

$$P_{ij}=\int_{T_j}^{1}\frac{1}{\sqrt{2\pi A_j}}e^{-\frac{(x-B_i)^2}{2A_j}}dx$$

When Li=0, Pij is:

$$P_{ij}=\int_{0}^{T_j}\frac{1}{\sqrt{2\pi A_j}}e^{-\frac{(x-B_i)^2}{2A_j}}dx$$

2.2.2 Process of Algorithm

The process of the EM algorithm is as follows:

1. Given L binary labels of M topic-document pairs annotated with by N workers, for each Worker W_j, initialize his ability value $1/A_j$ and save it.

2. E Step: For each pair D_i with N_i labels, calculate the expectation of its relevant value B_{emi} using the above Gaussian model and then save B_{emi}. B_{emi} is:

$$B_{emi}=\frac{\int_0^1 B_i\cdot\prod_{j=1}^{N_i}p_{ij}\,dB_i}{\int_0^1\prod_{j=1}^{N_i}p_{ij}\,dB_i}$$

3. M Step: For each worker W_j, estimate his ability value $1/A_j$ with maximum likelihood estimation and then save A_j.

4. Repeat the step 2-3 until all B_{emi} are stable or iterated enough times.

5. For each pair D_i, if B_{emi}>0.5, output label 1. Otherwise output label 0.

2.2.3 Program

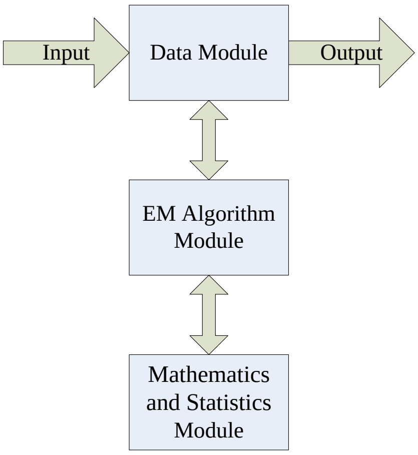


Figure 6 program

Its modules are as follows:

1. EM algorithm module: the implementation of the above EM algorithm.

2. Mathematics and Statistics module: using the open source math library GSL, to compute of expectation, integral and extreme value.

3. Data module: read the input data and write program result into the output file.

3. Result

The preliminary results are shown below, both task 1 and task 2. And in each task, results of both relevance and rank are given.

This is the result of binary judgement in task 1, compared with the average level of all teams:

team	Accuracy	Recall	Precision	Specificity
BUPT-WILDCAT	75.7%	83.8%	76.3%	64.2%
average	74.0%	75.4%	79.1%	70.4%

This is the result of rank judgement in task 1, compared with the average level of all teams:

team	MAP	NDCG
BUPT-WILDCAT	78.3%	82.0%
average	79.8%	83.1%

This is the result of binary judgement in task 2, compared with the average level of all teams:

team	Accuracy	Recall	Precision	Specificity
BUPT-WILDCAT	94.1%	92.3%	98.3%	97.2%
average	76.9%	80.0%	82.5%	71.6%

This is the result of rank judgement in task 2, compared with the average level of all teams:

team	MAP	NDCG
BUPT-WILDCAT	81.6%	92.8%
average	81.6%	92.1%

4. Conclusion and Future Work

In TREC, we made some adjustment in the strategies and algorithms, and the preliminary result shows that two-third of the criteria are high above the average level, especially in task 2. Our work provides an effective algorithm of consensus computing and a new method of assessment for the other researchers to refer. In future study, we will continue focus on improving the strategies and obtain a better result.

Acknowledgement

Thanks to the organizer for all the work and efficient communication. Thanks to Amazon.com and crowdflower.com for the generous financial and technical support. And we appreciate the help from other researchers in PRIS Laboratory of BUPT.

Reference

[1] TREC2011CrowdsourcingTrack-FinalGuidelines081811.pdf

[2] Dawid, A. P., and Skene, A. M. Maximum likelihood estimation of observer error-rates using the EM algorithm. Applied Statistics 28, 1 (Sept. 1979), 20-28.

[3] Panagiotis G. Ipeirotis, Foster Provost, Jing Wang. Quality Management on Amazon Mechanical Turk. KDD-HCOMP'10, July 25, 2010.

[4] Jacob Whitehill, Paul Ruvolo, Tingfan Wu, Jacob Bergsma, and Javier Movellan. Whose Vote Should Count More: Optimal Integration of Labels from Labelers of Unknown Expertise. Advances in Neural Information Processing Systems (forthcoming), 2009.