Wu *et al. EURASIP Journal on Wireless Communications and Networking* (2020) 2020:82
https://doi.org/10.1186/s13638-020-01697-2

EURASIP Journal on Wireless Communications and Networking

RESEARCH Open Access

Learning deep networks with crowdsourcing for relevance evaluation

Ming Wu[1], Xiaochun Yin[2], Qianmu Li[3,4*], Jing Zhang[1], Xinqi Feng[5], Qi Cao[6] and Haiyuan Shen[7]

Abstract

In this paper, we propose a novel relevance evaluation method using labels collected from crowdsourcing. The proposed method not only predicts the relevance between query texts and responses in information retrieval systems but also performs the label aggregation tasks simultaneously. It first merges two kinds of heterogeneous data (i.e., image and query text) and constructs a CNN-like deep neural network. Then, on the top of its softmax layer, an additional layer was built to model the crowd workers. Finally, classification models for relevance prediction and aggregated labels for training examples can be simultaneously learned from noisy labels. Experimental results show that the proposed method significantly outperforms other state-of-the-art methods on a real-world dataset.

Keywords: Crowdsourcing, Relevance evaluation, Information retrieval, Deep learning

1 Introduction

Relevance evaluation is a significant component in the domain of information retrieval [1–3] to develop and maintain IR systems, where relevance between queries and responses is an important indicators to reflect whether an IR system is good or not. Generally, the accuracy and relevance of IR systems could be improved furtherly using the feedback of relevance evaluation. In early years of the IR field, relevance evaluation tasks are usually performed by professional assessors or domain experts, but it has some limitations in practice. First, it is rather difficult for assessors to read a large number of documents and judge their relevance to corresponding query texts. Secondly, the process of evaluation is slow and expensive to insure the accuracy of judgments [4–6].

In 2006, the term *crowdsourcing* was first coined by Jeff Howe in the Wired magazine [7], and then Merriam-Webster defines crowdsourcing as the process of obtaining needed services, ideas, or content by soliciting contributions from a large group of people, and especially from an online community, rather than from traditional employees or suppliers. With the rapid development of crowdsourcing, many crowdsourcing systems have been generated, such as Amazon Mechanical Turk, CrowdFlower, and CloudCrowd [8–11]. Thanks to the growth of crowdsourcing platforms, we have an opportunity to improve the performance of relevance evaluation through crowdsourcing techniques because crowdsourcing provides a fast and low-cost solution to get numerous labeled data in near-real time from a vast of online Internet users [12–15]. By crowdsourcing platforms, requesters who have large tasks can split the tasks into plenty of small subtasks that common people without expertise can process, and then distribute the subtasks to tens of thousands of online workers. Finally, the responses collected from the online workers can be integrated into solutions of original tasks. In recent years, crowdsourcing has attracted lots of attentions from the domain of machine learning. As one of the important branches of machine learning, supervised learning performs well and steady and is widely used in many situations. Typically, supervised learning depends on amount of labeled data to train a model, so it is appropriate for researchers to obtain the data using crowdsourcing. Crowdsourcing provides a convenient

*Correspondence: qianmu@njust.edu.cn
[3]School of Cyber Science and Engineering, Nanjing University of Science and Technology, 200 Xiaolingwei Street, 210094 Nanjing, People's Republic of China
[4]Intelligent Manufacturing Department, Wuyi University, 529020 Jiangmen, People's Republic of China
Full list of author information is available at the end of the article

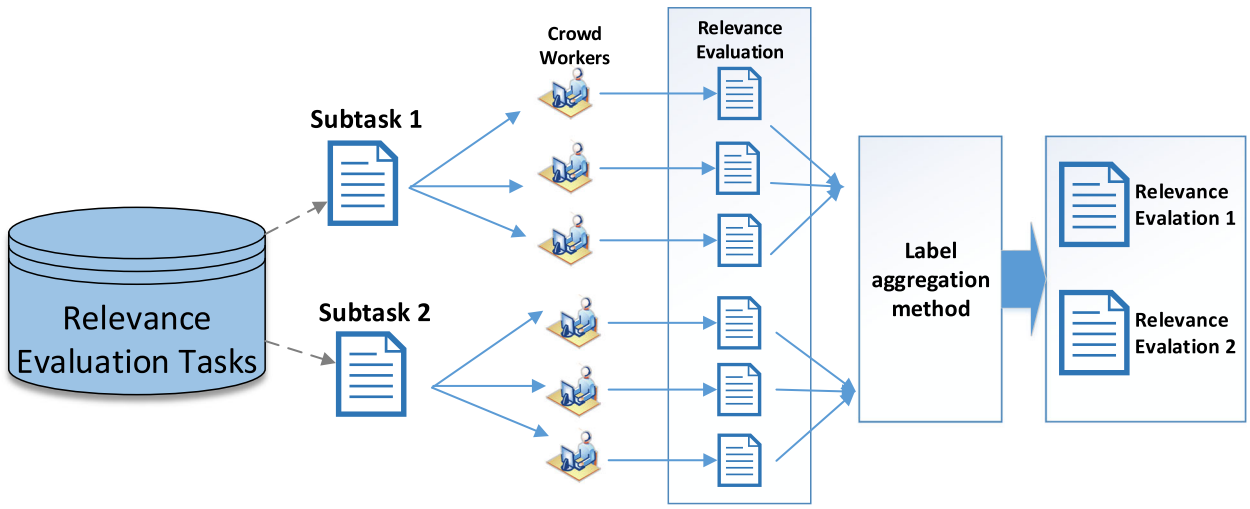

Fig. 1 Relevance evaluation. The processes of relevance evaluation tasks with crowd-workers

and low cost solution to get a large number of labeled data. Furthermore, annotations labeled by non-experts have proven to be reliable and effective [16].

However, relevance evaluation using crowdsourcing still faces challenges. In general, we assume that the labels provided by domain experts are correct and can be directly used to train a model, but the qualities of labels processed by crowdsourcing are varied. The reason is that the number of online workers is huge and they have great different contributions, professional abilities, and evaluation criterion in different tasks. In order to improve the quality of noisy data, a good way is to obtain redundant labels for each sample of the data; as repeated labeling can improve the qualities of labels and models, it is preferable to single labeling. After collecting more than one label for each sample, we can use several algorithms to infer an integrated label from the redundant labels for each sample, and these algorithms are called ground truth inference algorithms. We think that the integrated labels can be considered as substitutes for the ground truth of data, and then we can use these integrated labels to train a model. In addition, many researchers also do studies of building learning models directly using noisy data without processing the inference step first. To learn with noisy labels, researchers need to design stable models to address the affects of label noises. However, it is difficult and experiments show that the models trained using noise-tolerant methods are still influenced by noisy labels except for some simple cases. Crowdsourcing can be used to process many tasks, such as collecting ranking scores, labeling images and videos. Moreover, relevance evaluation is also a popular application of crowdsourcing. As we all know, relevance evaluation is a hard and expensive task, so crowdsourcing can perform well in this application. The process of relevance evaluation tasks with crowdworkers is described in Fig. 1. The relevance evaluation task can be divided into amounts of subtasks, and each subtask is assigned to multiple crowdworkers and labeled by the workers, then the multiple labels are aggregated into an integrated label as the ground truth. On the other hand, learning the features of workers is also an important and interesting topic in the field of crowdsourcing because we can utilize the information to select appropriate workers on specific tasks and dismiss spam and unreliable workers.

In this paper, we propose a novel method to process relevance evaluation using noisy labels collected by crowdsourcing with a deep learning architecture. In our method, relevance classification model can be learned directly from noisy labels by training a deep learning model. Furthermore, the trained deep learning model can aggregate the noisy labels to infer the ground truth, and it can also predict the relevance of new data, which further improve the efficiency of relevance evaluation tasks.

2 Related work

In this paper, we propose a novel ground truth inference and prediction method on the field of relevance evaluation by crowdsourcing. Generally, in the field of crowdsourcing, we can improve the qualities of labels by repeated labeling each sample and obtain the integrated labels. We think these integrated labels are appropriate substitutions for the hidden ground truth of the samples. After that, we can use supervised machine learning algorithms to train the data with the integrated labels.

To infer the integrated labels, there are many researches on inference algorithms. Majority voting (MV) is a naive and widely used method. In short, the integrated label

is the label provided by the majority of labelers. However, the MV model is too simple and it assumes that each labeler has the same ability to process the labeling tasks. David and Skene proposed an EM-based algorithm (expectation-maximization algorithm) called DS to infer the ground truth early in 1979 [17]. DS not only infer the integrated labels of samples, but also estimate a confusion matrix for each worker, and the confusion matrix can represent the reliability of the corresponding worker on each category. It can improve the accuracy of inference results. In addition, we can use the confusion matrices trained by DS to screen the workers. Although DS performs well in many situations [16, 18, 19], it has a limitation—if the number of categories is large, and the labels we collect is not enough relatively, then the confusion matrix will be sparse, which leads to incorrect results. Except for the accuracy of workers, the difficulty of each sample is also a useful factor. GLAD (Generative model of Labels, Abilities, and Difficulties) proposed a probabilistic model to estimate the label of each sample, the expertise of each worker, and the difficulty of each sample simultaneously [20]. Moreover, algorithms based on EM methods are not robust because of the defects of EM, since the likelihood function of EM is not convex, which means EM algorithms cannot converge to global optimal. To address this problem, a spectral method is utilized to estimate the initial values of the confusion matrix [21], which method is called Opt-D&S. Opt-D&S improves the accuracy than DS and shows that it achieves the optimal convergence rate.

Beside the algorithms based on EM methods, there are also several methods based on simple statistics or linear algebra. For example, GTIC (Ground Truth Inference using Clustering) is a statistics based algorithm proposed in [22]. GTIC is a ground truth inference algorithm used to solve multi-class labeling problems. If the example in dataset have K categories to distinguish, GTIC will run a clustering algorithm first on the dataset to divide the examples into K clusters. After that, GTIC will map each cluster to a specific category. Generally, deep learning is also considered to used in ground truth inferencing in several works [23–28]. Albarqouni et al. [23] provides an CNN-like network called AggNet to model the crowdworkers and inference process. AggNet learns multiple CNN models with same structures to model the capabilities of crowdworkers, and the outputs are the labels provided by the workers. The labels are then passed to an aggregation CNN to obtain the integrated label as the ground truth.

3 The proposed method

3.1 Preliminaries and motivations

The framework of our method is shown in Fig. 2. After crowdworkers provide their responses, we use all these labels to train a deep learning model, where the general structure of the model is shown in Fig. 3. Once the model is trained, the integrated labels can be obtained on the output layer of the model. Meanwhile, when a new task is input to the model, the relevance result will be predicted.

In this paper, we only discuss the evaluation of image search engine, so we define the entire data set as $\mathcal{D} = \{e_i\}_{i=1}^{I}$, and each example pair e_i is defined as $e_i =< q_i, p_i, y_i, \mathbf{l}_i >$, where q_i denotes the query text of example e_i, p_i denotes the image linked to the query text q_i, which means we obtain image p_i when we use text q_i to query in the search engine, notice that the linking do not imply that the image is relevant to the query since the search

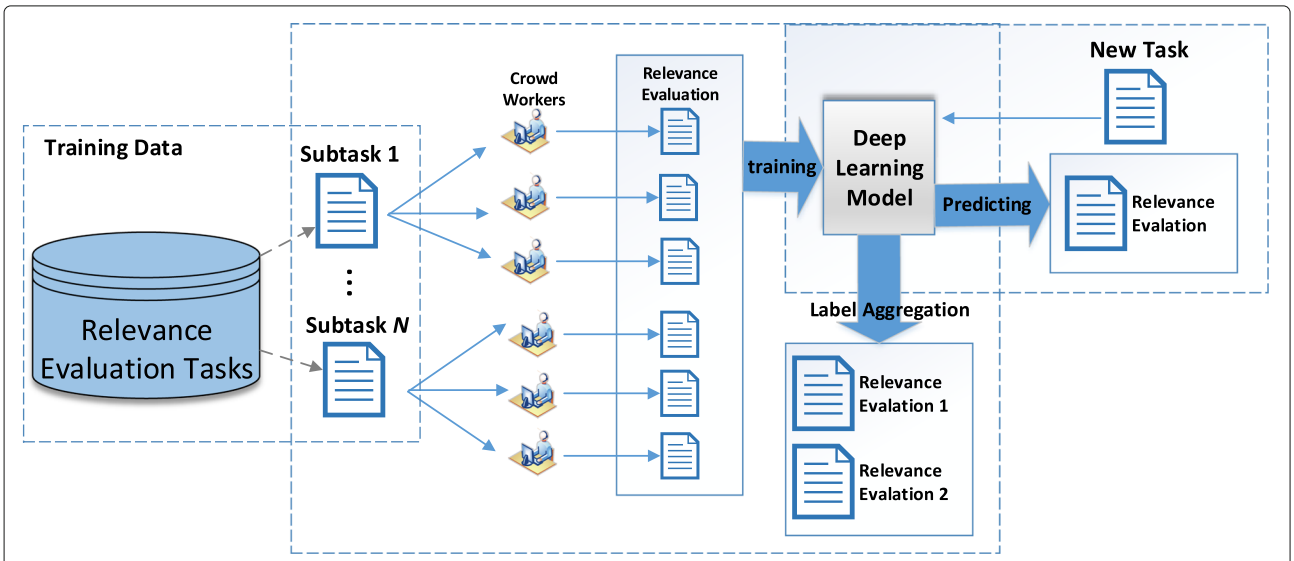

Fig. 2 The proposed framework. A framework of relevance evaluation with deep learning in crowdsourcing

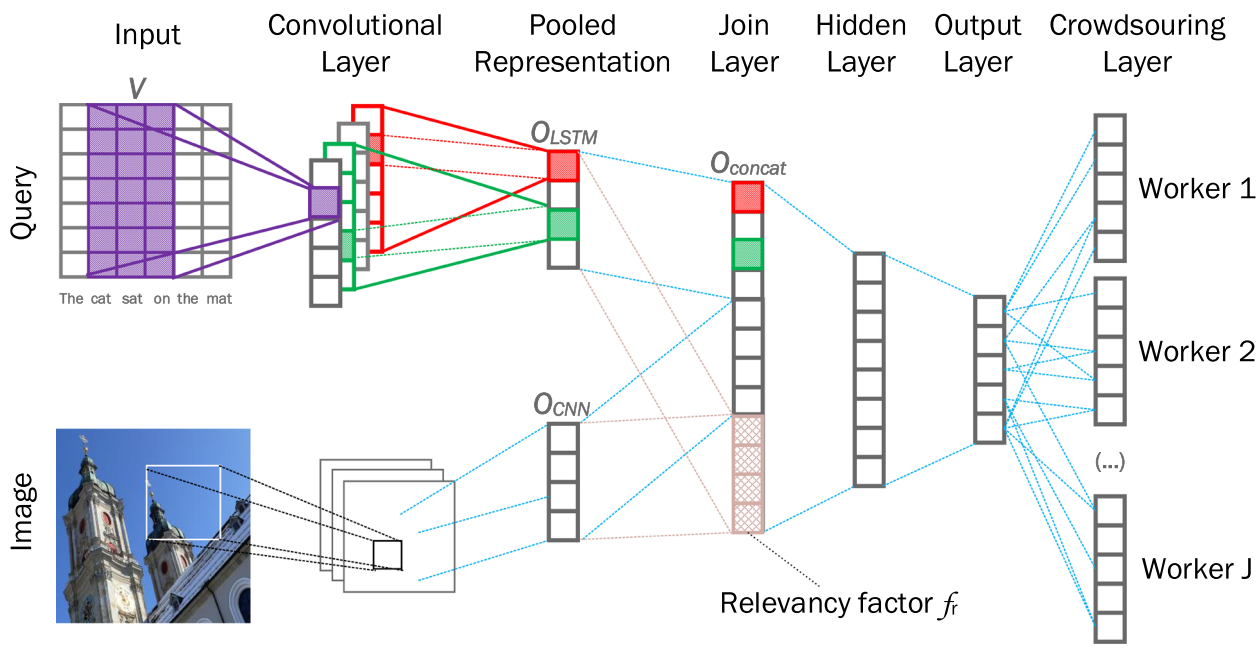

Fig. 3 Genaral structure. A general structure of our proposed method for relevance evaluation using deep learning, which assumes that the dataset contains 5 categories and J workers

engine sometimes do not return the right results, y_i means the ground truth of example e_i and contains two elements {relevance,irrelevance}, and $\mathbf{l}_i$ means the noisy label set of e_i labeled by multiple workers. In addition, the dataset of workers is defined as $\mathcal{W} = \{w_j\}_{j=1}^{J}$, so the noisy labels of example e_i can be defined as $\mathbf{l}_i = \{l_{ij}\}_{j=1}^{J}$, where l_{ij} denotes the label of example e_i provided by worker w_j. We also define the category set as $\mathcal{C} = \{c_k\}_{k=1}^{K}$; each label is selected in the set $\mathcal{C}$. In the case of relevance evaluation, we can simply map c_1 as relevant class and c_2 as irrelevant class, or we can define some more fine-grained categories.

3.2 The feature fusion deep learning method

In this section, we will introduce our method for relevance evaluation by crowdsourcing using a deep neural network. In social media photo retrieval tasks, we can search for the relevant photos by keywords or descriptive sentences or phrases on image-related websites such as Flickr. So, crowdsourcing will be an appropriate solution to evaluate the performance of search engines in a fast and low-cost way. In general, we distribute the subtasks of evaluation task to a large number of workers on crowdsourcing platforms after collecting the results of search engines, and the workers will judge the relevance of query-image pairs.

To estimate the relevance between images and texts, firstly, we need to extract the features, and deep learning is an appropriate choice to find the key features of data automatically. In deep learning, CNN is one of the neural networks, which is inspired by the visual principles of human brain, so CNN is commonly applied to image processing. In our case, we use a standard CNN architecture to learn the representations of images. So, a pre-trained model VGG-16 [29] is used to exact the features of images in our model, which is a CNN model with 13 convolutional layers and 3 fully connected layers trained on ImageNet database. Since deep learning models such as CNN comprise lots of hyperparameters, to learn the optimum of the hyperparameters is hardware-costing and time-consuming. One of the best methods is to improve the model base on the design and structure of professional teams. In this paper, we use the pre-trained model to reduce the training time and improve the accuracy of the model.

Here, we need to do a fine tuning on pre-trained VGG-16 model to learn the features that are more relevant to our own task. Since we do not have the ground truth to do the fine tuning, we use an autoencoder to work on it. Autoencoder is an unsupervised learning technique, which is a neural network that its output is same as its input, and we can use autoencoder to do the task of representation learning. We reconstruct the VGG-16 model as Fig. 4.

On the other hand, we firstly use the word2vec model trained with wiki corpus to obtain the word vectors of all the words in query texts $[v_1, ..., v_n]$, where n is the number of words in a query text. Word2vec model is a two-layer neural network used to produce word vectors,

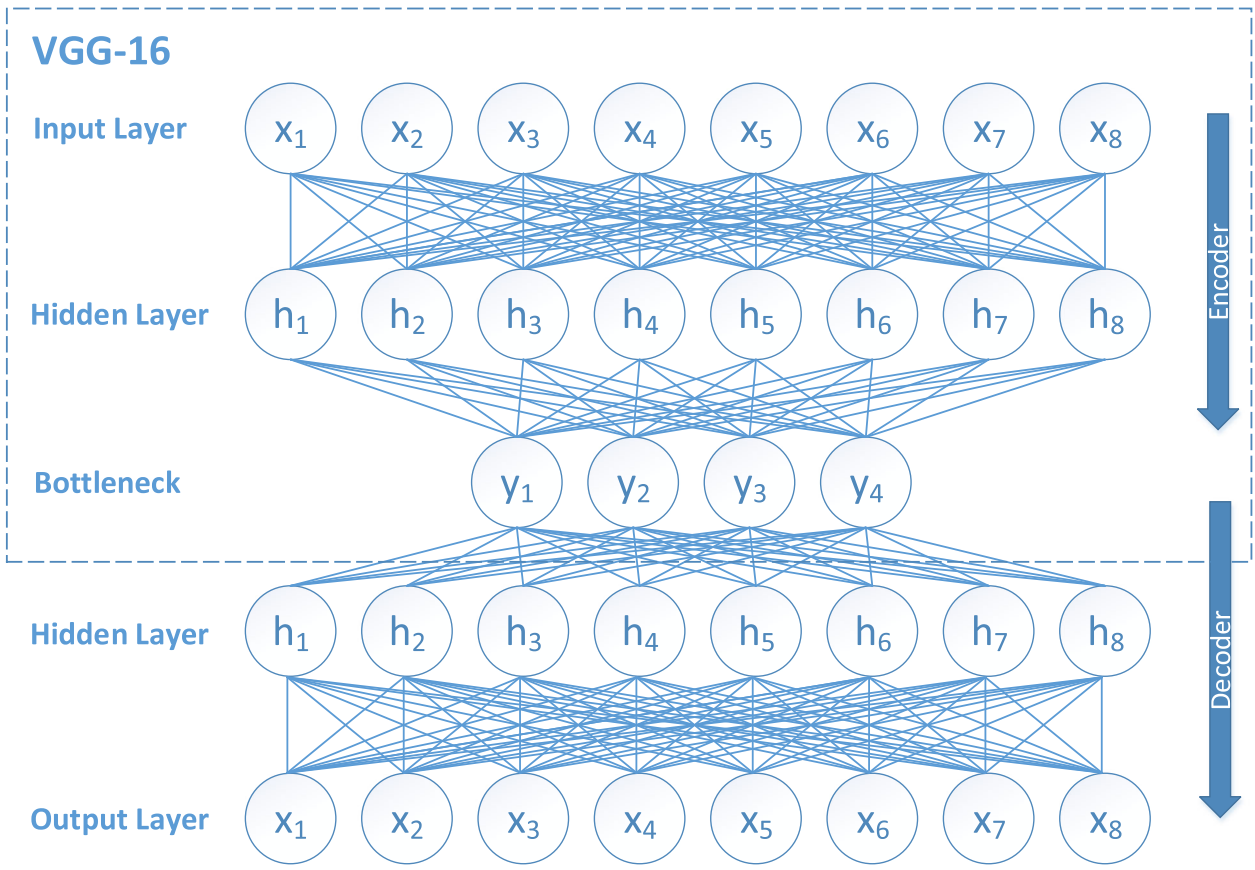

Fig. 4 Fine tuning. The reconstruction of VGG-16 to make an autoencoder for fine-tuning

which reflects semantic meanings of words and is useful in NLP tasks [30, 31]. After we obtain the word vectors, we can make a matrix $V \in \mathbb{R}^{n\times|v|}$ for each query text, and a LSTM network will be trained to learn the representations of the texts, where the input to the LSTM model is matrix V. In this step, LSTM is an RNN architecture used in deep learning, which has feedback connections to process time series data.

Moreover, we have to merge the two outputs produced from CNN and LSTM separately together. In general, many methods concatenate the two representations directly in this step as:

$$\mathbf{O}_{\text{concat}} = \left[\mathbf{O}_{\text{CNN}}^{T}; \mathbf{O}_{\text{LSTM}}^{T}\right] \tag{1}$$

where $\mathbf{O}_{\text{concat}}$ denotes the output of concatenate layer and $\mathbf{O}_{\text{CNN}}$ and $\mathbf{O}_{\text{LSTM}}$ denote the outputs of CNN and LSTM model respectively. However, most of the methods usually focus on the isomorphism data, such as that the query and response are both images or texts, so the concatenate feature can represent the fusion of two representations and performs well. Since the image and text data in our method are heterogeneous, the simple concatenate method cannot fuse the data well, so we need a more effective method to do the feature fusion. Since we want to mine the internal relevance of data deeply and use a parameter to measure the relevance, a similarity matrix $\mathbf{M}_r$ is proposed in this paper. We add a custom layer called *SMLayer* before the merge step to get a relevancy factor. The matrix $\mathbf{M}_r$ is a parameter of the model, which can be learned by training model, and the relevancy factor f_r which measures the relevance can be obtained as

$$f_r = \mathbf{O}_{\text{CNN}}^{T}\mathbf{M}_r\mathbf{O}_{\text{LSTM}} \tag{2}$$

The relevancy factor is widely used as a scoring model in IR as a machine translation. By combining the output of SMLayer f_r with the two representations, we can obtain a new feature fusion representation:

$$\mathbf{O}_{\text{concat}} = \left[\mathbf{O}_{\text{CNN}}^{T}; f_r; \mathbf{O}_{\text{LSTM}}^{T}\right] \tag{3}$$

Next, we concatenate two features and the factor, the output of concatenate layer is then input 5 fully connected (FC) layers orderly with ReLU activation function

to reduce the dimensions effectively and feed to an output layer with softmax activation. The output of softmax layer θ shows the result of classification, e.g., θ_{c_k} means the probability that the input example belongs to class c_k. To avoid overfitting, we use dropout to improve regularization, which can improve the performance of neural network by preventing the coefficient of feature detectors [32–35]. We apply 50% dropout between concatenate layer and the first fully connected layer. At last, a crowdsourcing layer is added on the top of softmax layer. With the crowdsourcing layer, we can model the disagreement of workers and map the output of softmax layer to the responses provided by workers, then all the noisy labels can be used in the deep network model without aggregation. The other configurations are selected from a set of possible options.

Figure 5 shows the detailed structure of our deep learning network. The network is trained using backpropagation, and the crowdsourcing layer can find out the unreliable workers and adjust their bias. In the crowdsourcing layer, there is a matrix transformation as $f(o) = \mathbf{M}^j o$, where $f(*)$ is the transformation function of crowdsourcing layer, o denotes the output of softmax layer, and $\mathbf{M}^j$ is a specific matrix of worker w_j. We may view $\mathbf{M}$ as the confusion matrix of each worker. The activation function of crowdsourcing layer can be seen as a softmax function. The model can be optimized by Adam and use logcosh as the loss function. Adam is an adaptive learning rate optimization algorithm which is popular for training deep neural networks, and it calculates individual learning rates for different parameters; the convergence rate of Adam is rapid and it performs well in practice. Also, logcosh is a common loss function and defined as $L(y, y^p) = \sum_{i=1}^{n} log\left(cosh\left(y_i^p - y_i\right)\right)$, it is not easy to be affected by outliers. Once the model is trained, the crowdsourcing layer could be removed and the remaining part can be used as a standard classifier. Obviously, the parameters $\mathbf{M}_j^J$ trained in the crowdsourcing layer reflects the reliability of each worker on different classes, and they can help to adjust the bias of workers. Finally, this model can not only obtain aggregated labels, but also predict relevance.

4 Experiments

4.1 Experimental setup

Our method was implemented in Keras [36], which is a high-level neural network API, and CEKA [37], which is an open software package for machine learning in crowdsourcing and contains many existing benchmarks for label aggregation. One real-world dataset was used in our experiments. Data set *Div400* is a image retrieval dataset [38], which was created to help evaluation in various areas of social media photo retrieval, such as re-ranking, crowdsourcing and relevance feedback. This data set gathered 15,871 Flickr photos and collected relevance evaluation by crowds, which contains 160 query texts and each query corresponding to less than 120 images.

Since there are some query texts have few responses and all of the responses belongs to the same category, we remove the query texts with all positive or negative responses. In addition, the data set is unbalanced, the number of negative examples is obviously far less than the number of positive examples; here, we utilize data argumentation to generate images when the negative examples is less than 10% of positive examples for each query. Since deep learning is a data-driven technology, a large amount of data is necessary to train an appropriate model, so data argumentation is used to increase the number of data manually using information only in our training data to avoid overfitting problems. The common data argumentation methods contain random crop, flip, rotate, resize and so on. Finally, we obtain 16,536 photos as training set and 1,170 photos as testing set. For each query and text pair, we simulate 5 workers to provide a label to evaluate whether this pair is relevant or irrelevant; let 1 and 0 denote relevant and irrelevant respectively. We give these workers different sensitivities μ and specificities γ to denote the different accuracies of workers on different categories because workers reflect different bias tendency towards the positive and the negative labels; when the ground truth of examples adapt to the bias of a worker, it would be a higher accuracy for the worker to provide true labels, and vice versa. For instance, worker j prefers to provide positive labels, so j would have higher accuracy to label positive examples. When the true label of a pair is 1, the worker j will provide the correct label with probability μ_j; otherwise, when the true label is 0, the probability will be γ_j. Afterwards, the values of parameters for 5 workers are $\mu = [0.6, 0.9, 0.5, 0.9, 0.9]$ and $\gamma = [0.3, 0.2, 0.5, 0.8, 0.1]$. To obtain the features of images, we use VGG-16 model pre-trained on ImageNet dataset. Also, for word embedding, we use word2vec trained on the English Wikipedia dump. In this paper, we use Keras to build a standard LSTM layer to train the text features. The outputs of LSTM and VGG-16 are reshaped the size into $(n, 384)$. The similarity matrix are set as $\mathbf{M}_r \in \mathbb{R}^{384\times384}$. The dimension of the output layer is 2 to represent the result is relevant or irrelevant.

We compare our method with five common ground truth inference methods MV, GTIC [22], Opt-D&S [21], DS [17], and GLAD [20].

The purpose of ground truth inference algorithms is to achieve the minimum of the empirical risk, then we will have a good chance of using integrated label $\hat{y}_i$ as the ground truth y_i.

$$\mathcal{R}_{emp} = \frac{1}{I}\sum_{i=1}^{I}\mathbb{I}\left(\hat{y}_i \neq y_i\right) \tag{4}$$

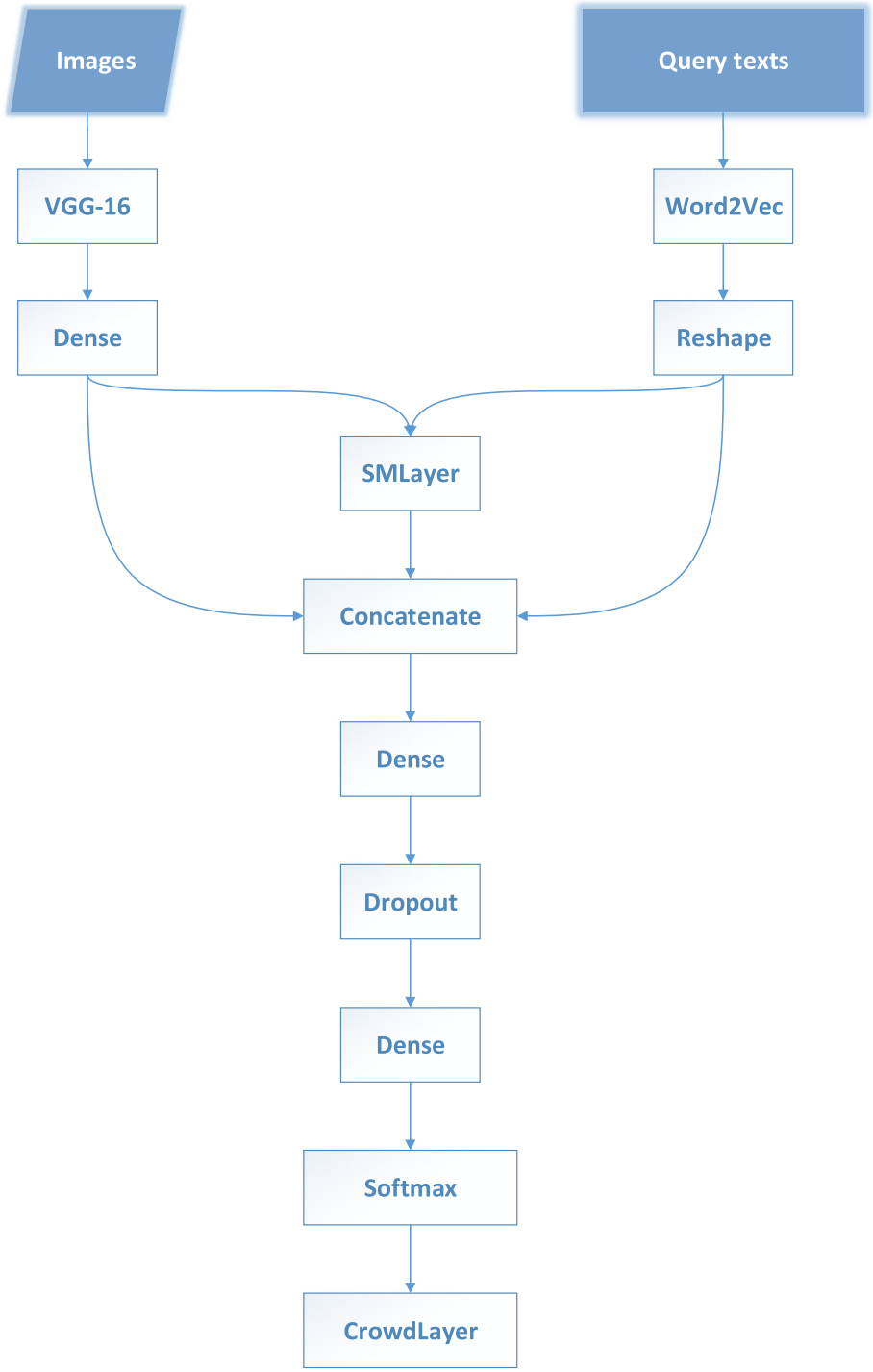

Fig. 5 Genaral structure. Structure of our proposed network for relevance evaluation using deep learning, which contains a crowd layer at the top of softmax layer

where $\mathbb{I}$ denotes an indicator function whose output will be 1 if the input is true, or else the output will be 0. As the simplest but widely used method, MV follows a simple principle that if more than half of the workers provide the same label c_k, then the integrated label will be c_k. DS is an EM-based method, it defined that each worker have an exclusive matrix $T^{(j)} = \left\{\pi_{kt}^{(j)}\right\}$ to indicate their labeling behavior called confusion matrix, where each element $\pi_{kt}^{(j)}$ denotes the probability that worker w_j provides label c_l to the example whose true label is c_k. The purpose of DS is to estimate the label of each example and the confusion matrix of each worker simultaneously. DS method contains two steps; firstly, we will initialize the confusion matrices and the prior probabilities of each category. Usually, we assume that all the parameters subject to uniform distribution, so $\pi_{kt}^{(j)} = \frac{1}{K}$ and $P(c_k) = \frac{1}{K}$, where $1 \leqslant k \leqslant K, 1 \leqslant t \leqslant K$, and $1 \leqslant j \leqslant J$. Then in E-step, it estimates the probability that each sample e_i belongs to each category c_k, that is

$$P\left(\hat{y}_i = c_k|\mathcal{D}\right) = \frac{\prod_{j=1}^{J}\prod_{t=1}^{K}\pi_{kt}^{(j)}P\left(c_k\right)}{\sum_{q=1}^{K}\prod_{j=1}^{J}\prod_{t=1}^{K}\pi_{qt}^{(j)}P\left(c_q\right)} \tag{5}$$

After that, in the M-step, we recalculate each confusion matrix and the prior probability of each category, that is

$$\hat{\pi}_{kt}^{(j)} = \frac{\sum_{i=1}^{I}\mathbb{I}\left(\hat{y}_i = c_k\right)}{\sum_{t=1}^{K}\sum_{i=1}^{I}\mathbb{I}\left(\hat{y}_i = c_k\right)} \tag{6}$$

$$\hat{P}\left(c_k\right) = \frac{1}{I}\sum_{i=1}^{I}\mathbb{I}\left(\hat{y}_i = c_k\right) \tag{7}$$

Afterwards, we can repeat the E-step and M-step until all the estimates converge. In addition, GLAD is also an EM-based method. Unlike DS method, GLAD not only considers about the ability of workers, but also takes the specificity of samples into consideration. GLAD models' two parameters α_j and β_i respectively denote the expertise of workers and the difficulty of samples, where $\alpha_j \in (-\infty, +\infty)$ and $1/\beta_i \in [0, +\infty)$. Worker w_j is more likely to provide a correct label when α_j is higher, when $\alpha_j = 0$, it means the worker w_j is making a wild guess. Similarly, sample e_i will be harder to be labeled correctly when $1/\beta_i$ is higher; when $1/\beta_i = 0$, it means that sample e_i is too easy to be classified that anyone can label it correctly. There, GLAD defines that

$$P\left(l_{ij} = y_i|\alpha_j, \beta_i\right) = \frac{1}{1 + e^{-\alpha_j\beta_i}} \tag{8}$$

Just like DS, after initializing the missing data, GLAD have two steps to iterative processing. In E-step, we obtain the posterior probabilities of all y_i:

$$P\left(y_i|\mathbf{l}_i, \alpha, \beta\right) \propto P\left(y_i\right)\prod_{j=1}^{J}P\left(l_{ij}|y_i, \alpha_j, \beta_i\right) \tag{9}$$

In the M-step, we can update the values of α and β by maximizing a standard auxiliary function $\mathcal{Q}$:

$$\begin{aligned}\mathcal{Q}\left(\alpha, \beta\right) &= E\left[lnP\left(\mathbf{l}, y|\alpha, \beta\right)\right] \\ &= \sum_{i}E\left[lnP(y_i)\right] + \sum_{ij}\left[lnP\left(l_{ij}|y_i, \alpha_j, \beta_i\right)\right]\end{aligned} \tag{10}$$

To avoid the limitation that EM-based algorithms cannot converge to the global optimal, Opt-D&S uses a spectral method to initialize the values of confusion matrix in DS method, which first divides the workers into three disjoint groups, and the average confusion matrix of the three groups is calculated separately. Then, the initial values of confusion matrix of each worker is set as the average value. In the second step, a DS algorithm runs with the initial values. Meanwhile, to solve the multi-class inference problems, GTIC [22] was proposed using Bayesian statistics. GTIC first generates K+1 features for each example e_i as $\boldsymbol{\alpha}^i = \left\{\alpha_1^i, ..., \alpha_{k+1}^i\right\}$. Secondly, any clustering algorithm will be used to run on the dataset and divide the examples into K clusters, the number of each cluster represented as $\mathbf{N}_n$, where $n = 1, ..., K$. Finally, for each cluster, we calculate a vector with k elements like:

$$\nu_k^n = \sum_{i}^{\mathbf{N}_n}\alpha_k^i, \text{ where } 1 \leqslant n \leqslant K \tag{11}$$

So, the maximum element ν_k^n in the vector ν^n will map the cluster n to the class k and all the examples belong to cluster n will be assigned to class k.

4.2 Experimental results

We first discuss the improvement of the model using similarity matrix. We build a base model which directly joint two representations in the merge step, then we introduce the similarity matrix into the model and compare the accuracy of two models [39, 40].

Figure 6 shows the comparison results of two models mentioned above, we find that on different configurations, the model using similarity matrix performs better than base model. With different configurations, the accuracies of model using similarity matrix are all higher than 62%, even higher than 70% when the batch size changes. On the other side, the model using concatenate are lower than

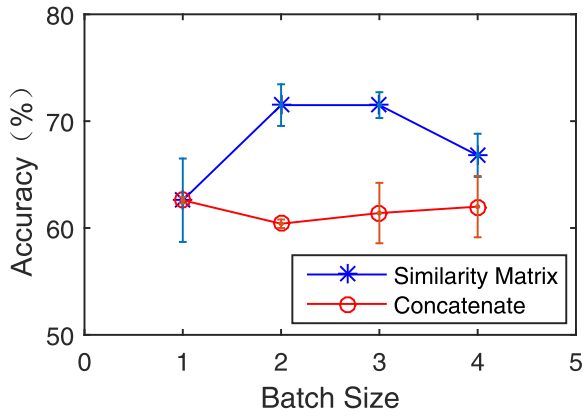

Fig. 6 Model comparison. Comparison results between base model and the model using similarity matrix

62% It demonstrates the effectiveness of similarity matrix; on the other hand, it shows that simply joint two representations to fuse the features cannot represent the data well in our case.

We investigate the effectiveness of our proposed method. Figure 7 shows the comparison results for aggregation. In our proposed method, we obtain the inference labels after the model is trained and remove the crowdsourcing layer.

As Fig. 7 shows, our method outperforms other five inference methods and the accuracy is 93.6%. We can find that the accuracy of our method is improved because we not only use the noisy labels to infer the ground truth, but also take the features of data into consideration. Besides, MV is still a robust method; the accuracy of MV is 84.4%, and it is the same as Opt-D&S and DS. GLAD also performs well with the accuracy 90.9%.

Therefore, in our third experiment, we show the prediction ability of our method. To do the comparison experiments, we use MV, GTIC, Opt-D&S, DS, and GLAD respectively to infer the integrated labels of training data, then we use these labels to train a base deep learning model without crowdsourcing layer. Meanwhile, we also use the ground truth to train the model [41, 42]. The comparison results are shown in Fig. 8.

As shown in Fig. 8, the accuracy of our method is 73.9%, which is much higher than others. The model trained by ground truth also performs well with the accuracy of 71.2%. The results demonstrate the effectiveness for use of crowd information.

5 Discussion

In this paper, a novel relevance evaluation method was proposed with a deep learning network using crowdsourcing data to improve the accuracy compared with traditional methods and furtherly reduce the cost and time. We suggest a novel research direction to work on relevance evaluation without or reducing the involvement of human. We train a deep learning network to figure out the relevance between query texts and images directly from crowdsourcing data end-to-end.

Generally, to obtain the relevance between image and text, there are some other technologies such as *Image Caption* [43–45], which generate a sentence or several words to describe the content of image automatically, and then compare with the text. However, we will lose some

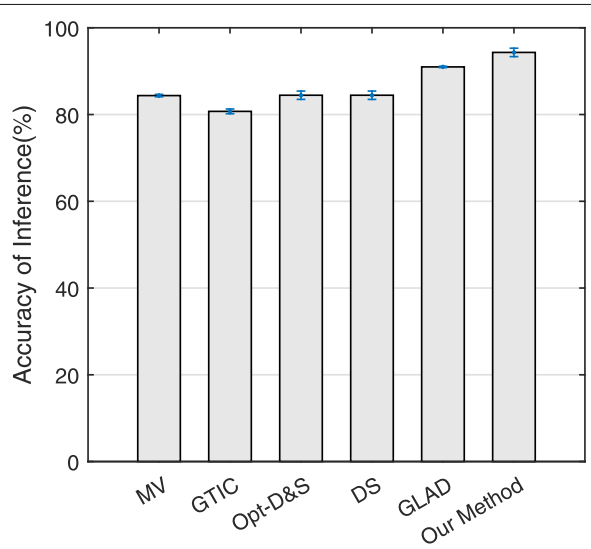

Fig. 7 Accuracy comparison. Comparison results in accuracy for six aggregation methods

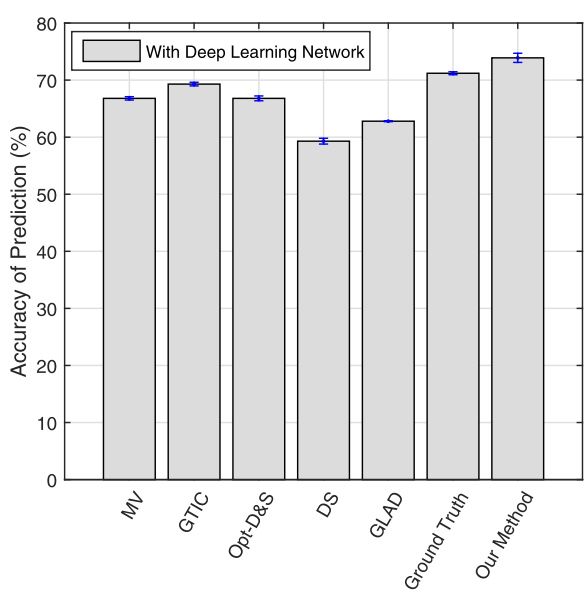

Fig. 8 Prediction comparison. Comparison results of prediction models trained by six methods

information when we generate the image caption, so it is preferable to directly compare image and text. Meanwhile, another way is to match the categories of images and text, which firstly classify the images and texts separately, and then compare the category of image and text. However, this way, can just judge whether the image and text belong to the same category, but cannot figure out the relevance fine-grainly.

In this paper, we use one dataset to show the performance of our method on images and texts, so we suggest that our method can be applied on multi-modal field in the future, which can also operate on video, speech, and other types of data. Also, more effective deep learning network structures should be studied to adapt to different application scenarios.

vsectionConclusions The proposed relevance evaluation method using crowdsourcing labels can effectively improve the accuracy of both aggregating and predicting. We take the features of data into consideration. Furthermore, the inference methods may lose information of the workers' characteristic; in our method, we can keep the information of every worker and train the model end to end by using deep learning technology with a crowdsourcing layer. Experimental results on a real-world dataset show that the proposed method outperforms other state-of-the-art methods in aggregation and prediction.

Abbreviations

CNN: Convolutional neural networks; IR: Information retrieval; NLP: Natural language processing; LSTM: Long short-term memory; RNN: Recurrent neural network; ReLU: Rectified linear unit; MV: Majority vote; EM: Expectation-maximization

Acknowledgements

Not applicable.

Authors' contributions

MW is responsible for carrying out the key design and implementation of the proposed model, preprocessing of the data set, and initialization of the drafted manuscript. XY and QL contributed to the conception and design of the study and also contributed the experimental evaluation sections in the manuscript. JZ and XF took main responsibilities to the analysis and discussion of the experimental results and also commented on the work and contributed to the final version of manuscript. All authors read and approved the final manuscript.

Funding

This work was supported in part by the Fundamental Research Funds for the Central Universities (30918012204), Military Common Information System Equipment Pre-research Special Technology Project (315075701), 2019 Industrial Internet Innovation and Development Project from Ministry of Industry and Information Technology of China, 2018 Jiangsu Province Major Technical Research Project "Information Security Simulation System", Shanghai Aerospace Science and Technology Innovation Fund (SAST2018-103), and National Natural Science Foundation of China (91846104).

Availability of data and materials

Not applicable.

Competing interests

We declare that all authors have no significant competing financial, professional, or personal interests that might have influenced the performance or presentation of the work described in this manuscript.

Author details

[1]School of Computer Science and Engineering, Nanjing University of Science and Technology, 200 Xiaolingwei Street, 210094 Nanjing, P.R.China. [2]Facility Horticulture Laboratory of Universities in Shandong, WeiFang University of Science & Technology, ShouGuang, People's Republic of China. [3]School of Cyber Science and Engineering, Nanjing University of Science and Technology, 200 Xiaolingwei Street, 210094 Nanjing, People's Republic of China. [4]Intelligent Manufacturing Department, Wuyi University, 529020 Jiangmen, People's Republic of China. [5]Center Of Informationization Construction And Management, Nanjing Sport Institute, 8 Linggusi Street, 210094 Nanjing, People's Republic of China. [6]Academy of Science and Technology Strategic Consulting, Chinese Academy of Science, 100190 Beijing, People's Republic of China. [7]Jiangsu Zhongtian Internet Technology Co, Ltd., 226463 Nantong, People's Republic of China.

Received: 6 January 2020 Accepted: 4 April 2020

References

1. O. Alonso, D. E. Rose, B. Stewart, in *ACM SigIR Forum, vol. 42*. Crowdsourcing for relevance evaluation (ACM, 2008), pp. 9–15
2. C. Kong, G. Luo, L. Tian, X. Cao, Disseminating authorized content via data analysis in opportunistic social networks. Big Data Min. Analytics. **2**(1), 12–24 (2018)
3. S. Kumar, M. Singh, Big data analytics for healthcare industry: impact, applications, and tools. Big Data Min. Analytics. **2**(1), 48–57 (2018)
4. B. Wang, H. Ma, X. Wang, G. Deng, Y. Yang, S. Wan, Vulnerability assessment method for cyber-physical system considering node heterogeneity. J. Supercomput. **75**(10), 1–21 (2019). https://doi.org/10.1007/s11227-019-03027-w
5. Y. Wang, Q. He, D. Ye, Y. Yang, Formulating criticality-based cost-effective fault tolerance strategies for multi-tenant service-based systems. IEEE Trans. Softw. Eng. **44**(3), 291–307 (2017)
6. Q. He, R. Zhou, X. Zhang, Y. Wang, D. Ye, F. Chen, J. C. Grundy, Y. Yang, Keyword search for building service-based systems. IEEE Trans. Softw. Eng. **43**(7), 658–674 (2016)
7. J. Howe, The rise of crowdsourcing. Wired Mag. **14**(6), 1–4 (2006)
8. L. Qi, Y. Chen, Y. Yuan, S. Fu, X. Zhang, X. Xu, A QOS-aware virtual machine scheduling method for energy conservation in cloud-based cyber-physical systems. World Wide Web. **23**(1), 1275–1297 (2020). https://doi.org/10.1007/s11280-019-00684-y
9. Q. He, J. Han, F. Chen, Y. Wang, R. Vasa, Y. Yang, H. Jin, in *2015 IEEE 8th International Conference on Cloud Computing*. QOS-aware service selection for customisable multi-tenant service-based systems: Maturity and approaches (IEEE, 2015), pp. 237–244
10. Y. Xu, L. Qi, W. Dou, J. Yu, Privacy-preserving and scalable service recommendation based on simhash in a distributed cloud environment. Complexity. **2017**, 1–9 (2017)
11. X. Xu, R. Mo, F. Dai, W. Lin, S. Wan, W. Dou, Dynamic resource provisioning with fault tolerance for data-intensive meteorological workflows in cloud. IEEE Trans. Ind. Inform. **2019**, 1–1 (2019)
12. M. Lease, E. Yilmaz, Crowdsourcing for information retrieval. ACM SIGIR Forum. **45**(2), 66–75 (2012)
13. Z. Junlong, S. Jin, C. Peijin, L. Zhe, W. Tongquan, Z. Xiumin, H. Shiyan, Security-critical energy-aware task scheduling for heterogeneous real-time MPSoCs in IoT. IEEE Trans. Serv. Comput. (TSC) (2019). https://doi.org/10.1109/TSC.2019.2963301
14. X. Xu, Y. Xue, L. Qi, Y. Yuan, X. Zhang, T. Umer, S. Wan, An edge computing-enabled computation offloading method with privacy preservation for internet of connected vehicles. Futur. Gener. Comput. Syst. **96**, 89–100 (2019)
15. P. Lai, Q. He, G. Cui, X. Xia, M. Abdelrazek, F. Chen, J. Hosking, J. Grundy, Y. Yang, Edge user allocation with dynamic quality of service. International Conference on Service-Oriented Computing, 86–101 (2019)
16. R. Snow, B. O'Connor, D. Jurafsky, A. Y. Ng, Cheap and fast—but is it good?: evaluating non-expert annotations for natural language tasks (Association for Computational Linguistics). *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, 254–263 (2008)
17. A. P. Dawid, A. M. Skene, Maximum likelihood estimation of observer error-rates using the em algorithm. Appl. Stat. **28**(1), 20–28 (1979)

18. J. Zhou, X. S. Hu, Y. Ma, J. Sun, T. Wei, S. Hu, Improving availability of multicore real-time systems suffering both permanent and transient faults. IEEE Trans. Comput. **68**(12), 1785–1801 (2019)
19. J. Zhou, J. Sun, X. Zhou, T. Wei, M. Chen, S. Hu, X. S. Hu, Resource management for improving soft-error and lifetime reliability of real-time MPSoCs. IEEE Trans. Comput. Aided Des. Integr. Circ. Syst. **38**(12), 2215–28 (2018)
20. J. Whitehill, T.-f. Wu, J. Bergsma, J. R. Movellan, P. L. Ruvolo, Whose vote should count more: optimal integration of labels from labelers of unknown expertise. *Advances in Neural Information Processing Systems*. **22**, 2035–2043 (2009)
21. Y. Zhang, X. Chen, D. Zhou, M. I. Jordan, Spectral methods meet em: a provably optimal algorithm for crowdsourcing. J. Mach. Learn. Res. **17**(1), 3537–3580 (2016)
22. J. Zhang, V. S. Sheng, J. Wu, X. Wu, Multi-class ground truth inference in crowdsourcing with clustering. IEEE Trans. Knowl. Data Eng. **28**(4), 1080–1085 (2016)
23. S. Albarqouni, C. Baur, F. Achilles, V. Belagiannis, S. Demirci, N. Navab, Aggnet: deep learning from crowds for mitosis detection in breast cancer histology images. IEEE Trans. Med. Imaging. **35**(5), 1313–1321 (2016)
24. S. Wan, S. Goudos, Faster R-CNN for multi-class fruit detection using a robotic vision system. Comput. Netw. **168**, 1–16 (2019)
25. S. Wan, L. Qi, X. Xu, C. Tong, Z. Gu, Deep learning models for real-time human activity recognition with smartphones. Mob. Networks Appl. **75**(12), 1–13 (2019). https://doi.org/10.1007/s11036-019-01445-x
26. Z. Gao, H.-Z. Xuan, H. Zhang, S. Wan, K.-K. R. Choo, Adaptive fusion and category-level dictionary learning model for multi-view human action recognition. IEEE Internet Things J. **6**, 9280–9293 (2019)
27. Z. Gao, Y. Li, S. Wan, Exploring deep learning for view-based 3D model retrieval. ACM Trans. Multimed. Comput. Commun. Appl. (TOMM). **16**(1), 1–21 (2020)
28. Y. Zhao, H. Li, S. Wan, A. Sekuboyina, X. Hu, G. Tetteh, M. Piraud, B. Menze, Knowledge-aided convolutional neural network for small organ segmentation. IEEE J. Biomed. Health Inform. **23**(4), 1363–1373 (2019)
29. K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556. **2014**, 1–14 (2014)
30. T. Mikolov, K. Chen, G. Corrado, J. Dean, Efficient estimation of word representations in vector space. arXiv preprint arXiv:1301.3781. **2014**, 1–12 (2013)
31. X. Rong, word2vec parameter learning explained. arXiv preprint arXiv:1411.2738. **2014**, 1–21 (2014)
32. S. Wan, X. Li, Y. Xue, W. Lin, X. Xu, Efficient computation offloading for Internet of Vehicles in edge computing-assisted 5G networks. J. Supercomput. **75**(10), 1–30 (2019). https://doi.org/10.1007/s11227-019-03011-4
33. S. Wan, Z. Gu, Q. Ni, Cognitive computing and wireless communications on the edge for healthcare service robots. Comput. Commun. **149**, 99–106 (2019)
34. S. Wan, Y. Xia, L. Qi, Y.-H. Yang, M. Atiquzzaman, Automated colorization of a grayscale image with seed points propagation. IEEE Trans. Multimed. **2020**, 1–1 (2020)
35. S. Ding, S. Qu, Y. Xi, S. Wan, Stimulus-driven and concept-driven analysis for image caption generation. Neurocomputing. **351**, 1–10 (2019)
36. F. Chollet, et al., Keras (2015). https://github.com/fchollet/keras. 10 Accessed 13 June 2015
37. J. Zhang, V. S. Sheng, B. A. Nicholson, X. Wu, Ceka: a tool for mining the wisdom of crowds. J. Mach. Learn. Res. **16**(1), 2853–2858 (2015)
38. B. Ionescu, A.-L. Radu, M. Menéndez, H. Müller, A. Popescu, B. Loni, Div400: a social image retrieval result diversification dataset. *Proceedings of the 5th ACM Multimedia Systems Conference*, 29–34 (2014)
39. X. Xu, X. Liu, Z. Xu, C. Wang, S. Wan, X. Yang, Joint optimization of resource utilization and load balance with privacy preservation for edge services in 5G networks. Mob. Netw. Appl., 1–12 (2019)
40. X. Xu, C. He, Z. Xu, L. Qi, S. Wan, M. Z. A. Bhuiyan, Joint optimization of offloading utility and privacy for edge computing enabled IoT. IEEE Internet Things J. **2019**, 1–1 (2019)
41. H. Liu, H. Kou, C. Yan, L. Qi, Link prediction in paper citation network to construct paper correlation graph. EURASIP J. Wirel. Commun. Netw. **2019**(1), 1–12 (2019)
42. L. Qi, Q. He, F. Chen, W. Dou, S. Wan, X. Zhang, X. Xu, Finding all you need: Web APIs recommendation in web of things through keywords search. IEEE Trans. Comput. Soc. Syst. **6**(5), 1063–1072 (2019)
43. O. Vinyals, A. Toshev, S. Bengio, D. Erhan, Show and tell: a neural image caption generator. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 3156–3164 (2015)
44. W. Gong, L. Qi, Y. Xu, Privacy-aware multidimensional mobile service quality prediction and recommendation in distributed fog environment. Wirel. Commun. Mob. Comput. **2018**, 1–8 (2018)
45. L. Qi, X. Zhang, W. Dou, C. Hu, C. Yang, J. Chen, A two-stage locality-sensitive hashing based approach for privacy-preserving mobile service recommendation in cross-platform edge environment. Futur. Gener. Comput. Syst. **88**, 636–643 (2018)

Publisher's Note