scientific reports

OPEN

Deep active learning for multi label text classification

Qunbo Wang[1,4], Hangu Zhang[2,4], Wentao Zhang[2], Lin Dai[2], Yu Liang[3] & Haobin Shi[2]✉

Given a set of labels, multi-label text classification (MLTC) aims to assign multiple relevant labels for a text. Recently, deep learning models get inspiring results in MLTC. Training a high-quality deep MLTC model typically demands large-scale labeled data. And comparing with annotations for single-label data samples, annotations for multi-label samples are typically more time-consuming and expensive. Active learning can enable a classification model to achieve optimal prediction performance using fewer labeled samples. Although active learning has been considered for deep learning models, there are few studies on active learning for deep multi-label classification models. In this work, for the deep MLTC model, we propose a deep Active Learning method based on Bayesian deep learning and Expected confidence (BEAL). It adopts Bayesian deep learning to derive the deep model's posterior predictive distribution and defines a new expected confidence-based acquisition function to select uncertain samples for deep MLTC model training. Moreover, we perform experiments with a BERT-based MLTC model, where BERT can achieve satisfactory performance by fine-tuning in various classification tasks. The results on benchmark datasets demonstrate that BEAL enables more efficient model training, allowing the deep model to achieve training convergence with fewer labeled samples.

Nowadays, deep learning model achieves the state-of-the-art result in the area of multi-label text classification. It is well known that deep learning model requires a large amount of labeled data for training, which limits their adoption in many real-world scenarios. Even with the advancement of crowd-sourcing systems[1], acquiring large-scale and high-quality human annotations remains costly and time-consuming[2]. Active learning[3] is a technology used to optimize the utilization of an annotation budget in such scenarios. The primary idea of active learning is to progressively choose and annotate most informative unlabeled samples for model learning[4]. It can often effectively reduce the annotation effort without compromising the model performance when training a learning model[5].

The previous research efforts of active learning are often designed for the traditional machine learning models, e.g. SVM[6,7]. These methods can't be directly used by deep learning models. They either fail to scale to high-dimensional data samples or depend on accurate uncertainty estimates for unlabeled data samples, which are challenging to obtain with standard neural networks[8]. Due to the high complexity of deep learning models, it is necessary to make efforts on designing the corresponding deep active learning methods. Recently, there have been more and more researches studying deep active learning. These research efforts are often designed for single-label classification, in which each data sample is associated with only one class label[8–15]. These methods are mostly not directly well applicable in multi-label classification, because sample selection decisions in multi-label classification should be based on all labels[16]. In other words, these related methods cannot be directly applied in the deep multi-label classification model to help reduce the required labeled training samples.

Compared to single-label classification, multi-label classification problem is more practical, as real-world objects are usually associated with multiple labels, such as topics or attributes[17]. In multi-label classification problem, each object can be associated with multiple labels simultaneously[18]. Therefore, the output space is exponentially larger than that of single-label classification, it usually requires a large amount of labeled data to train an effective deep multi-label classification model[19]. On the other hand, the data annotation process in a multi-label scenario is much more expensive and time-consuming compared to a single-label scenario. In the single-label scenario, annotators can stop labeling a sample once a correct label is found. But in the multi-label scenario, annotators need to judge all labels for each sample. Therefore, it is important to study active learning to minimize human annotation efforts for building deep multi-label classification models.

Bayesian formulations of neural networks provide alternative techniques for inferring the posterior predictive distribution of deep learning models[20]. And Gal et al.[21] demonstrated that performing stochastic forward passes with dropout is equivalent to approximate Bayesian inference in neural networks, which makes it easier to apply Bayesian methods to obtain the deep model's posterior predictive distribution. In this work, we focus on the

[1]Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China. [2]Northwestern Polytechnical University, Xi'an 710129, China. [3]Beijing University of Technology, Beijing 100124, China. [4]These authors contributed equally to this work: Qunbo Wang and Hangu Zhang. ✉email: shihaobin@nwpu.edu.cn

pool-based active learning and propose an active learning method called BEAL for the deep multi-label text classification model. BEAL utilizes dropout to implement Bayesian deep learning to infer the deep model's posterior predictive distribution. Using this distribution, it defines a new expected confidence-based acquisition function for multi-label classification, aiming to choose uncertain samples that the model lacks confidence to predict. Moreover, experiments are conducted with a BERT-based MLTC model. The results on the benchmark datasets including AAPD and StackOverflow reveal that BEAL enables more efficient model training, allowing the deep model to achieve training convergence with fewer labeled samples. In summary, the main contributions of the paper are as follows:

- We propose an active learning method for the deep multi-label text classification model, which defines a new expected confidence-based acquisition function. This method can be easily extended to the deep learning models for other multi-label classification tasks.
- We implement active learning on a BERT-based multi-label text classification model, where BERT-based models can achieve state-of-art performances by fine-tuning in many classification tasks. There is a lack of research efforts on applying active learning to a BERT-based multi-label classification model.
- The experimental results reveal that the proposed method can more effectively reduce the need of labeled training samples compared to other methods.

Related work

Active learning

In conventional supervised learning, a machine learning model is passively assigned a set of labeled data samples to be trained on. In contrast, active learning permits the learning model to interactively request supervision according to its own choice[22]. Active learning scenarios primarily include: membership query synthesis, stream-based selective sampling, and pool-based sampling[5]. In the first scenario, the learner generates synthetic samples in the input space, and requests labels for them. However, some of these generated samples might prove challenging to label in a reasonable way[23,24]. In the second scenario, data samples are continuously provided in a stream-like manner, and therefore decisions about whether an unlabeled sample should be labeled are made individually[25]. In the third scenario, there is a pool of unlabeled samples that is made available at the beginning. All samples from this unlabeled pool are evaluated before selecting which of them are to be labeled[22].

In our work, we focus on the pool-based active learning scenario, because it is suitable for a large number of real-world problems[5,26,27]. In the stream-based scenario, new data samples usually arrive in batches, which can be essentially handled using pool-based active learning methods. Therefore, it is reasonable to assume that our method can be applied to the stream-based scenario as well[22].

Active learning for multi-label classification

Automatic classification based on deep learning is very important in practical applications[28,29]. Active learning for multi-label classification has been widely studied[7,16,30–33], in which uncertainty-based sampling is the most frequently used acquisition strategy. For example, Li et al.[16] proposed two multi-label active learning strategies, a max-margin prediction uncertainty strategy and a label cardinality inconsistency strategy, and then integrated them into an adaptive framework of active learning. However, these methods are designed for traditional machine learning models.

In addition, aggregation of multiple labels' uncertainty score is studied by some researchers[22,32]. For example, Cherman et al.[22] proved that the deviation-based aggregation strategy achieves an advantageous performance in multi-label active learning. On the other hand, there are also some studies to exploit the label hierarchies for multi-label active learning. For example, Yan et al.[19] proposed a criterion to calculate the informativeness of each object-label pairs to exploit the label hierarchies. However, these methods only can be implemented where labels have a hierarchy.

Recently, based on the Bayesian theory, Goudjil et al.[7] proposed an active learning method for SVM classifiers. In this method, samples are acquired according to the posterior predictive distribution that is obtained by constructing an ensemble of SVM classifiers. This method often becomes computationally expensive and can't be directly adopted in deep learning models which need more computational cost for training. Therefore, it is necessary to design active learning methods for deep multi-label classification models.

Deep active learning

Although there are many researches on active learning, it is still difficult to apply these classic methods to deep learning because they are usually unable to handle high-dimensional data or measure the uncertainty of neural networks[34]. Deep learning requires a large amount of labeled data to train a massive number of parameters, and synergy between deep learning and active learning has received attention from the research community[8–15,35,36]. For example, Beluch et al.[13] trained an ensemble of neural networks to derive uncertainty estimates for unlabeled data samples; nevertheless, concurrently training such an ensemble of networks simultaneously often incurs substantial computational expenses. Sener et al.[12] proposed a density-based method called Core-Set to cover the entire feature space of unlabeled data using a geometric similarity function among samples, which can be implemented in different deep learning models. Ash et al.[15] proposed a method called BADGE to select samples that are disparate and of high magnitude when represented in a hallucinated gradient space, and proved its effectiveness across different network architectures. In our experiments, we choose Core-Set and BADGE as the comparison methods. Saran et al.[37] proposed an algorithm for batch active learning with deep neural networks in streaming settings. Yuan et al.[38] proposed a method selecting and annotating the unlabeled samples to train the deep CNN model. Mollenbrok et al.[35] introduced a deep active learning method for image classification.

Bayesian deep learning provides alternative techniques for inferring the posterior predictive distribution over neural networks[20,39,40]. And Gal et al.[21] demonstrated that performing stochastic forward passes with dropout is equivalent to approximate Bayesian inference, which makes it easier to apply Bayesian deep learning to derive the posterior distribution of deep model prediction. Further, Gal et al.[8] introduced an active learning framework based on Bayesian deep learning, and find that it has more advantages than other active learning methods in the image classification task.

Most research efforts of deep active learning focus on single-label classification tasks[8–15]. Currently, there are few studies on active learning for deep multi-label classification models. The multi-label classification task is more difficult than the single-label classification task, which presents more challenges to designing deep active learning methods. In our work, for the deep multi-label text classification model, we propose an active learning method based on Bayesian deep learning and the expected confidence.

Multi-label text classification model

The multi-label text classification task requires assigning multiple labels for a given text, in which the deep learning model can achieve a satisfying performance and is adopted in our work. Denotes X_i as an input text item, which includes a sequence of words. Let $D_{labeled} = \{(X_1, L_1), (X_2, L_2), ...\}$ denotes the labeled training dataset, in which $L_i \in \{0, 1\}^J$ is the ground-truth of the labels corresponding to X_i and *J* is the total number of labels. Specifically, $L_{ij} = 1$ if X_i belongs to the *jth* label and otherwise $L_{ij} = 0$. The target is to train a deep learning model using the labeled dataset $D_{labeled}$, which can predict the most relevant labels to the new text samples.

The basic architecture of the neural network is shown in Fig. 1. Similar to the related work[41], we implement a BERT-based model as our deep multi-label text classification model and fine-tune BERT during model training. In our model, all words of an input item X_i are tokenized and fed into BERT, which can output a sentence feature *C*. Because BERT limits the input length to 510 tokens, according to the common approach, the input text with more than 510 tokens will be truncated from the right end. Then, the sentence feature *C* is input to a hidden layer and an output layer with the sigmoid activation. Last, the model outputs a prediction value Y_{ij} for each label, which belongs to [0, 1]. And if Y_{ij} is greater than 0.5, the *jth* label is predicted to be relevant to the text. Otherwise, the *jth* label can be classified as being unrelated to the text.

Methodology

In this section, we describe the active learning framework in our work firstly. Then, we introduce a new data acquisition function.

Active learning framework

Many traditional active learning methods only acquire one sample from the unlabeled data pool in each round. However, deep learning is often very greedy for labeled data, where a single data sample can't bring a significant impact on model learning[34]. In contrast, batch-mode active learning acquires a batch of samples for labeling in each round, making it more appropriate for models with slow training procedures or parallel labeling environments[5]. Therefore, our work adopts batch-mode active learning as it facilitates the effective training of deep learning models.

Within our pool-based batch-mode active learning framework, we commence with a limited set of labeled samples $D_{labeled} = \{(X_1, L_1), (X_2, L_2), \dots, (X_k, L_k)\}$ to train the initial deep model and a large pool of unlabeled data samples $D_{unlabeled} = \{X_{k+1}, X_{k+2}, \dots\}$.The subsequent process involves conducting multiple rounds of data labeling and model retraining until the performance of the model's predictions converges. At the start of each round, a data acquisition function *A*() selects a batch of samples from $D_{unlabeled}$ for annotation.

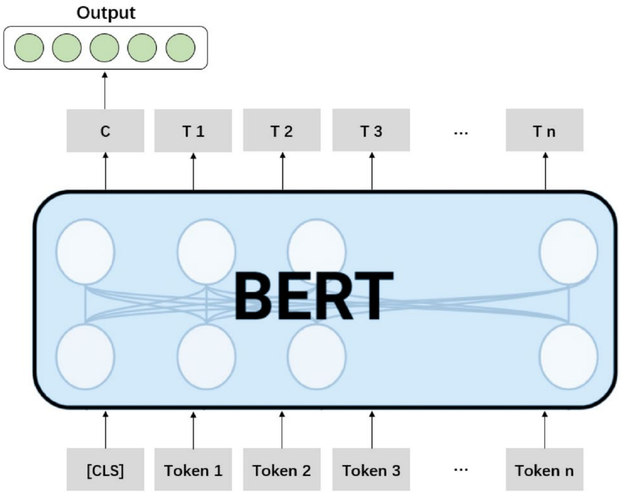


Fig. 1. The neural network architecture.

Then, these labeled data samples are added into $D_{labeled}$. Last, we update the parameters of the deep model by retraining it from scratch on the expanded labeled dataset $D_{labeled}$ and proceed to the next round.

The proposed acquisition function

One common and most used strategy in active learning is to acquire the uncertain samples for model learning[5], which can be measured by the model's predictive confidence. This section describes the proposed acquisition function in active learning, which selects samples with the lowest expected confidence.

For an unlabeled sample *X*, the deep multi-label text classification model (denoted as *M*) aims to assign the most relevant labels by outputting a prediction value $Y_j (j \in \{1, .., J\})$ for each label. Because the prediction value is generated by a squashing function 'sigmoid', we input it to a function *R*() and use the output as the relevance score predicted by the model *M*, which can better represent the relevance degree between *X* and the *jth* label.

$$R(Y_j) = 2^{Y_j} - 1 \tag{1}$$

Suppose there is a label list π sorted in descending order by the relevance score of each label. For any two adjacent labels in π, the preceding label is expected to be more relevant to the sample than the next label. If the model predicts a higher relevance score for an expected relevant label, it is obvious that the model is more confident. Therefore, for two adjacent labels in π, the relevance score of the preceding label is more important for determining the model's confidence.

To measure the confidence of a prediction *Y* in *M*, we propose a function as defined in Equation (2), in which the relevance scores of the labels are accumulated from the top of π to the bottom, with the score of each label discounted at lower ranks.

$$conf(\pi, Y) = \sum_{j}^{J} \frac{R(Y_j)}{\pi(j)} \tag{2}$$

where $\pi(j)$ is the position of the *jth* label in the sorted list π.

Further, to measure the model's confidence for the unlabeled sample *X*, the Bayesian framework is implemented to obtain the expected confidence:

$$EC(X) := \int_{Y} conf(\hat{\pi}, Y) p(Y|X, D_{labeled}) \mathrm{d}Y \tag{3}$$

where $p(Y|X, D_{labeled})$ is the posterior predictive distribution for *X*, and $\hat{\pi}$ is the label list sorted by the posterior relevance score of each label.

To obtain $p(Y|X, D_{labeled})$ of the deep model *M*, Bayesian deep learning[40] is adopted in our work. In Bayesian deep learning, the prediction of *M* is viewed as a probabilistic model $p(Y|X, \omega)$, in which prior probability distributions are assigned to the model parameters: $\omega \backsim p(\omega)$. To obtain $p(Y|X, D_{labeled})$, we need to calculate the posterior distribution of the model parameters ω given the labeled training data $D_{labeled}$:

$$p(\omega|D_{labeled}) = \frac{p(D_{labeled}|\omega)p(\omega)}{p(D_{labeled})} \tag{4}$$

And the posterior predictive distribution of the model *M* for the sample *X* can be formulated as follows:

$$p(Y|X, D_{labeled}) = \int_{\omega} p(Y|X, \omega) p(\omega|D_{labeled}) \mathrm{d}\omega. \tag{5}$$

However, in the above formulations, the posterior distribution $p(\omega|D_{labeled})$ cannot usually be evaluated analytically for neural networks. In practice, variational inference can be adopted to obtain an approximating distribution that minimises the Kullback-Leibler (KL) divergence with $p(\omega|D_{labeled})$[40]. And Gal and Ghahramani[21] demonstrated that performing stochastic forward passes with dropout is equivalent to variational inference in Bayesian deep learning, in which $p(\omega|D_{labeled})$ is approximated using the dropout distribution $q^*_\phi(\omega)$. Therefore, the posterior predictive distribution can be obtained by marginalising over $q^*_\phi(\omega)$ using Monte Carlo dropout sampling[21]:

$$\begin{aligned} p(Y|X, D_{labeled}) &\approx \int_{\omega} p(Y|X, \omega) q^*_\phi(\omega) \mathrm{d}\omega \\ &\approx \frac{1}{T} \sum_{t=1}^{T} p(Y|X, \hat{\omega}^t) \end{aligned} \tag{6}$$

where $\hat{\omega}^t$ denotes the model parameters over forward pass t with dropout and $\hat{\omega}^t \backsim q^*_\phi(\omega)$. In addition, for X, the prediction Y of the model M is determined over a forward pass. And for simplicity, we represent Y over forward pass t as Y^t. And we can obtain the approximated expected confidence as follows:

$$EC(X) \approx \frac{1}{T}\sum_{t=1}^{T} conf(\hat{\pi}, Y^t) \tag{7}$$

To obtain the label list $\hat{\pi}$, we sort the labels according to the approximated posterior relevance score of each label, which can be calculated as follows:

$$\frac{1}{T}\sum_{t=1}^{T} R(Y_j^t) \tag{8}$$

Last, we implement an acquisition function $A()$ based on the expected confidence in our active learning framework:

$$A(D_{unlabeled}, M) = \underset{X \in D_{unlabeled}}{\arg\min} \; EC(X) \tag{9}$$

In each round of active learning, the function $A()$ is used to select b samples with the lowest expected confidence for labeling, where b is the query batch size. The details of our proposed active learning method are given in Algorithm 1.

```
Input:
    Initially labeled dataset $D_{labeled}$.
    The unlabeled data pool $D_{unlabeled}$.
    Query batch size $b$.
    Maximum round number $N$ in active learning.
Output:
Initialize the model parameters $\omega$ with $D_{labeled}$
while not reach maximum round $N$ do
    for each unlabeled data sample $X \in D_{unlabeled}$ do
        for forward pass with dropout $t = 1, ..., T$ do
            The deep MLTC model outputs a prediction score $Y_j^t$ for each label over forward pass $t$
        end for
        for $j = 1, ..., J$ do
            Calculate the approximated posterior relevance score of the $jth$ label: $\frac{1}{T}\sum_{t=1}^{T} R(Y_j^t)$
        end for
        Obtain $\hat{\pi}$ sorted by the approximated posterior relevance score of each label
        Calculate $EC(X)$ following Eq. (7)
    end for
    Label $b$ samples with the lowest $EC$ value and add them into $D_{labeled}$
    Retrain the model parameters $\omega$ with $D_{labeled}$ from scratch
end while
return The model parameters $\omega$.
```

Algorithm 1. BEAL algorithm.

Experiments

Experimental setup

In this section, we present the datasets used, comparison methods, and implementation details.

Datasets

The proposed method is evaluated with two benchmark multi-label text classification datasets, including the dataset AAPD[42] and the real-world dataset StackOverflow (https://stackoverflow.com). The details of the data used in our experiments are summarized in Table 1.

Dataset	N_{train}	N_{test}	*L*
AAPD	10000	1267	54
StackOverflow	10000	1386	41

Table 1. The statistics of the datasets in our experiments. N_{train} is the number of training samples used for active learning, N_{test} is the number of test samples, *L* is the total number of labels.

Comparison methods

To illustrate the superior effectiveness of our proposed method BEAL for the deep multi-label text classification model, we compare it with the most representative deep active learning methods that can be implemented in multi-label text classification.

Random sampling This method acquires samples uniformly at random from the unlabeled data pool.

Batch active learning by diverse gradient embeddings (BADGE) This method acquires the samples that are disparate and high magnitude when represented in a hallucinated gradient space, which can incorporate both predictive uncertainty and sample diversity into every selected batch[15].

Bayesian active learning by disagreement (BALD) BALD selects samples based on the uncertainty of the model's predictions. Utilizing the dropout approximation technique in Bayesian deep learning, this method involves choosing samples that exhibit the highest probability assigned to various labels in each stochastic forward pass with dropout[8].

Core-set This method chooses the samples that best cover the dataset in the learned representation space[12]. And we adopt a recently enhanced version[43].

All data In this method, we directly query the labels of all the training samples and use them to train the deep multi-label text classification model.

Implementation details

In our experiments, the model was firstly trained with the initially labeled dataset $D_{labeled}$ including 500 samples, which was a random subset from the training dataset, and the rest samples in the training dataset were used as the unlabeled data $D_{unlabeled}$ for the active learning process.

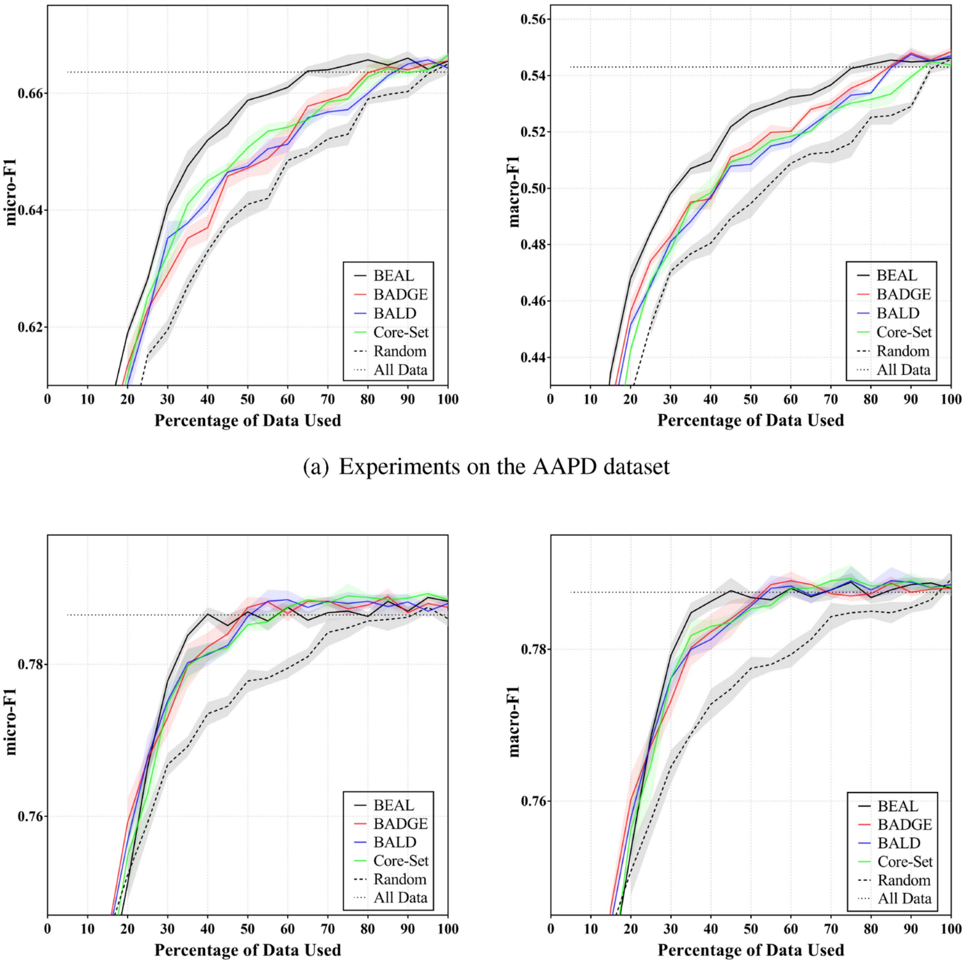
All the active learning methods were run for 19 acquisition rounds. In each acquisition round, the method acquired 500 samples from the unlabeled dataset $D_{unlabeled}$. Once the labels of these samples were retrieved, the deep multi-label text classification model was retrained from scratch using the augmented labeled dataset $D_{labeled}$ for a fixed number of epochs and evaluated using the test samples. Specifically, we adopted the standard metrics micro-F1 and macro-F1 in multi-label classification to evaluate the model prediction performance.

To reduce the randomness in the model training, the experiment was repeated five times for each active learning method. The experimental results provided the mean and standard deviation over five runs. For the active learning methods based on Bayesian deep learning, besides employing dropout in the model for training, we also utilized dropout to approximate the posterior predictive distribution including 100 forward passes. The implementation of the proposed method is based on Pytorch. All experiments were trained on GPU NVIDIA Tesla V100.

Results

To clearly evaluate the effectiveness of BEAL, we conduct experiments and discuss them in this section. Figure 2 illustrates the micro-F1 and macro-F1 curves of active learning methods under different percentages of labeled data used for training. Specifically, it provides the mean and standard deviation over five runs of each active learning method, where shaded areas in the figures denote standard deviation. The two groups of experiments use the AAPD and StackOverflow datasets, respectively. Note that because each sample is often crucial for model training at the initial stage, the performance discrepancy between various methods is minimal when the percentage of acquired samples stays less than 10%. Consequently, to adhere to the page limit, we cut off the curves at the initial stage to highlight the contrast between different methods at the remaining stages.

From Fig. 2, one can make the following observations about these results. Firstly, we observe the poor performance of Random Sampling in all settings. The reason is that it only selects samples uniformly at random, which fails to capture informative samples for model training. Secondly, we find that the performance of Core-Set, choosing the data samples that best cover the dataset, is not satisfying. It illustrates that it is not very effective to acquire diverse samples for model training in multi-label classification, where different samples may be associated with different numbers of labels and learning how to predict for each label is more important. Moreover, it is noteworthy that BALD's performance is also unsatisfactory, despite its strategy of acquiring samples with the highest probability assigned to various labels in each stochastic forward pass with dropout. This is because all the labels are predicted as positive with their prediction score greater than 0.5, thus hindering the BALD algorithm from accurately measuring sample uncertainty. Thirdly, BADGE performs better than the other baseline methods. It demonstrates that incorporating both predictive uncertainty and sample diversity into the acquisition function is helpful for BADGE to select the informative samples for model training. Nevertheless, it is clear that BEAL achieves the best in all the given settings. This is because BEAL computes the expected confidence of a data sample based on the posterior predictive distribution. It not only takes into account the fluctuation of the prediction but also computing the confidence by exploring the prediction score of each label simultaneously. Therefore, comparing with BADGE, BEAL can measure the uncertainty of each sample more accurately in multi-label classification, which enables BEAL to select the most informative samples without



(a) Experiments on the AAPD dataset

Fig. 2. The experimental results in different settings. It is obvious that our proposed method BEAL works better than the other deep active learning methods and Random Sampling in all settings.

Dataset	Metric	Core-Set	BALD	BADGE	BEAL
AAPD	micro-F1	83%	86%	81%	**64%**
	macro-F1	93%	85%	84%	**76%**
StackOverflow	micro-F1	57%	51%	49%	**40%**
	macro-F1	59%	54%	53%	**44%**

Table 2. The percentage of labeled samples required by different active learning methods to achieve the performance of all data. The best results are highlighted in bold.

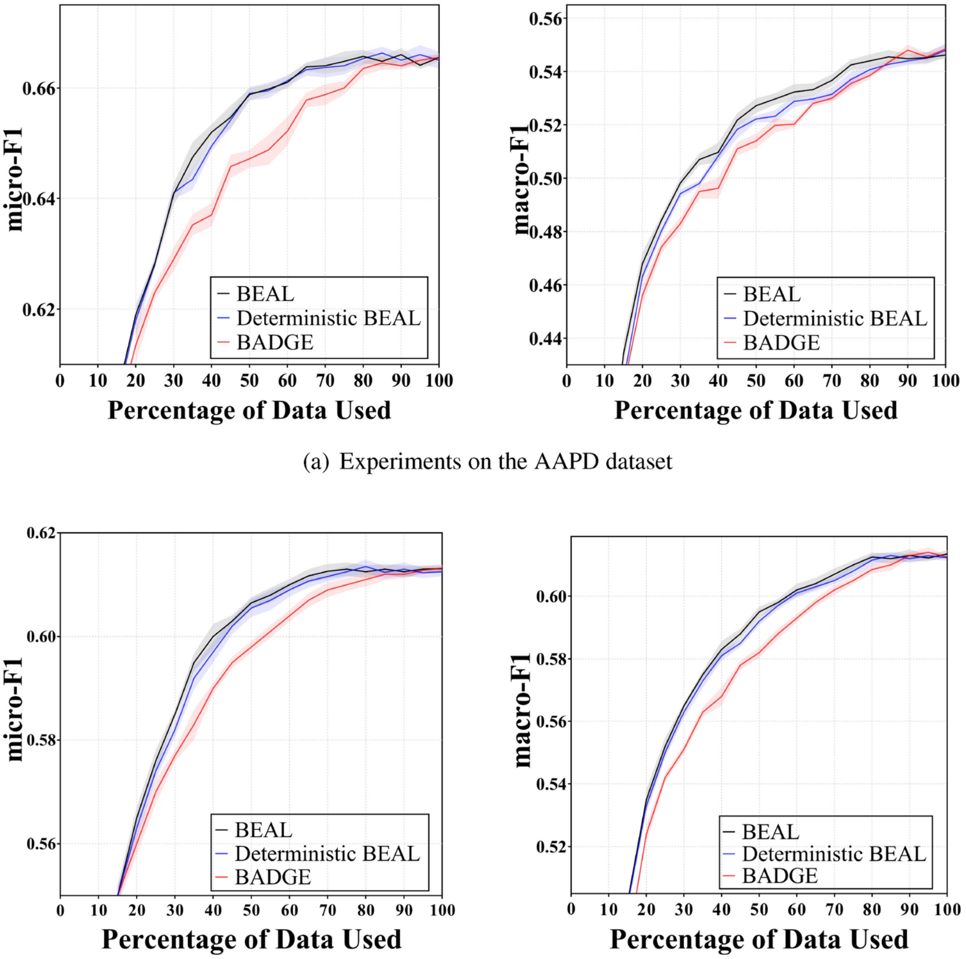
considering the sample diversity. This can let the model use the most beneficial samples for training to help improve the performance. Note that all the curves converge to the best performance because all samples are acquired for training at the end.

Specifically, Fig. 2a and b show the results on the AAPD dataset and the StackOverflow dataset respectively. Obviously, the best performance that the deep multi-label text classification model can achieve on the AAPD dataset is lower than that on the StackOverflow dataset, which illustrates that it is more difficult for the model to learn on the AAPD dataset. Hereby, it demands more labeled data for the same methods to converge to the best performance on the AAPD dataset. And the AAPD dataset can better test performance differences between active learning methods. As shown in Fig. 2, the gap between the curves of different methods is more obvious on the AAPD dataset.

Then, we compare the percentage of labeled data required by different active learning methods to achieve the performance by training with all data. As shown in Table 2, the advantages of BEAL over the other deep active learning methods can be substantial in all settings. For example, under the micro-F1 metric, BEAL requires only 64% and 40% labeled training samples to achieve the performance of All Data on the AAPD dataset and the StackOverflow dataset, respectively. As a comparison, BADGE requires 81% and 49% labeled training samples,

(a) Text: our paper explores contribution patterns of creativity and collaboration of wikipedia editors as manifestations of social dynamics between the editors we find support for existence of four socially constructed personas among the editors and difference in distribution of personas in articles of different qualities
Labels: cs.SI, physics.soc-ph
(b) Text: this paper presents a method for computing a least fixpoint of a system of equations over booleans the resulting computation can be significantly shorter than the result of iteratively evaluating the entire system until a fixpoint is reached
Labels: cs.PL, cs.SE

Table 3. An example of the data in AAPD, including two samples. Each sample consists of a text and its corresponding labels, where sample (b) is easier for the model to predict and has a lower learning value.



(a) Experiments on the AAPD dataset

(b) Experiments on the StackOverflow dataset

Fig. 3. Results of the ablation test.

BALD requires 86% and 51% labeled training samples, and Core-Set requires 83% and 57% labeled training samples. In summary, our proposed method BEAL effectively reduces the need of labeled training samples in comparison to other deep active learning methods. This is attributed to BEAL's ability to mine the most informative samples to provide sufficient training data for model learning.

Case studies

As shown in Table 3, for the model, Sample (b) is easier to predict than Sample (a). While Sample (b) clearly reflects its related topic, the topic of Sample (a) is harder to discern from its text, making it more uncertain for the model. If Sample (a) is manually labeled and used for model training, it can significantly enhance the model's knowledge. In other words, it helps the model learn more efficiently.

Ablation study

BEAL performs stochastic forward passes with dropout to implement Bayesian deep learning to infer the deep model's posterior predictive distribution. This section assesses the effect of Bayesian deep learning by comparing with a variant, **Deterministic BEAL**, which directly inputs a prediction of the deep multi-label classification text model to Eq. (2). Figure 3 gives the ablation results. We can observe that the performance of BEAL is better than Deterministic BEAL in Fig. 3a and b. The reason is that using the posterior predictive distribution enables the proposed acquisition function to better estimate the model's confidence. In addition, Deterministic BEAL is still better than the best baseline BADGE, which suggests the importance of the proposed confidence measure function Eq. (2).

Limitations

Our method performs stochastic forward passes with dropout to obtain an approximating distribution of the model prediction, which inevitably adds some computational overhead. However, since only forward propagation is performed without updating the model parameters, this extra cost remains manageable. Nonetheless, more efficient techniques for approximating the posterior predictive distribution can be explored in the future to further minimize this additional computational burden.

Conclusion

In this work, we propose an active learning method BEAL for the deep multi-label text classification model. It adopts Bayesian deep learning to infer the posterior distribution of model prediction and defines a new expected confidence-based acquisition function to select the uncertain samples for model training. The experimental results illustrate that BEAL outperforms other deep active learning methods in the benchmark MLTC datasets. It enables more efficient model training, allowing the deep model to achieve training convergence with fewer labeled samples. This can assist in the practical implementation of deep model, as obtaining sufficient labeled data is often challenging in real-world applications. In the future, we will study how to incorporate our expected confidence-based method and diversity-based method to further reduce the labeled date required for model training.

Data availability

The data are available from the corresponding author upon request.

Received: 7 June 2024; Accepted: 7 November 2024
Published online: 15 November 2024

References

1. Kittur, A., Chi, E. H. & Suh, B. Crowdsourcing user studies with mechanical turk. In *Proc. of the SIGCHI conference on human factors in computing systems*, 453–456 (ACM, 2008).
2. Wang, Q., Wu, W., Qi, Y. & Zhao, Y. Deep bayesian active learning for learning to rank: A case study in answer selection. *IEEE Trans. Knowl. Data Eng.* (2021).
3. Tharwat, A. & Schenck, W. A survey on active learning: state-of-the-art, practical challenges and research directions. *Mathematics* **11**, 820 (2023).
4. Cohn, D. A., Ghahramani, Z. & Jordan, M. I. Active learning with statistical models. *J. Artif. Intell. Res.* **4**, 129–145 (1996).
5. Settles, B. *Active learning literature survey*. Tech. Rep., University of Wisconsin-Madison Department of Computer Sciences (2009).
6. Brinker, K. On active learning in multi-label classification. In *From Data and Information Analysis to Knowledge Engineering*, 206–213 (Springer, 2006).
7. Goudjil, M., Koudil, M., Bedda, M. & Ghoggali, N. A novel active learning method using svm for text classification. *Int. J. Autom. Comput.* **15**, 290–298 (2018).
8. Gal, Y., Islam, R. & Ghahramani, Z. Deep bayesian active learning with image data. In *Proc. of the 34th International Conference on Machine Learning*, 1183–1192 (2017).
9. Wang, K., Zhang, D., Li, Y., Zhang, R. & Lin, L. Cost-effective active learning for deep image classification. *IEEE Trans. Circuits Syst. Video Technol.* **27**, 2591–2600 (2016).
10. Fang, M., Li, Y. & Cohn, T. Learning how to active learn: A deep reinforcement learning approach. Preprint at http://arxiv.org/abs/1708.02383 (2017).
11. Shen, Y., Yun, H., Lipton, Z. C., Kronrod, Y. & Anandkumar, A. Deep active learning for named entity recognition. Preprint at http://arxiv.org/abs/1707.05928 (2017).
12. Sener, O. & Savarese, S. Active learning for convolutional neural networks: A core-set approach. In *International Conference on Learning Representations* (2018).
13. Beluch, W. H., Genewein, T., Nürnberger, A. & Köhler, J. M. The power of ensembles for active learning in image classification. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition*, 9368–9377 (2018).
14. Siddhant, A. & Lipton, Z. C. Deep bayesian active learning for natural language processing: Results of a large-scale empirical study. Preprint at http://arxiv.org/abs/1808.05697 (2018).
15. Ash, J. T., Zhang, C., Krishnamurthy, A., Langford, J. & Agarwal, A. Deep batch active learning by diverse, uncertain gradient lower bounds. In *8th International Conference on Learning Representations* (2020).
16. Li, X. & Guo, Y. Active learning with multi-label svm classification. In *IJCAI*, 1479–1485 (Citeseer, 2013).
17. Zhu, F., Li, H., Ouyang, W., Yu, N. & Wang, X. Learning spatial regularization with image-level supervisions for multi-label image classification. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition*, 5513–5522 (2017).
18. Tsoumakas, G., Zhang, M.-L. & Zhou, Z.-H. Introduction to the special issue on learning from multi-label data (2012).
19. Yan, Y. & Huang, S.-J. Cost-effective active learning for hierarchical multi-label classification. In *IJCAI*, 2962–2968 (2018).
20. Blundell, C., Cornebise, J., Kavukcuoglu, K. & Wierstra, D. Weight uncertainty in neural networks. Preprint at http://arxiv.org/abs/1505.05424 (2015).
21. Gal, Y. & Ghahramani, Z. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *Proc. of the 33th International Conference on Machine Learning*, 1050–1059 (2016).
22. Cherman, E. A., Papanikolaou, Y., Tsoumakas, G. & Monard, M. C. Multi-label active learning: key issues and a novel query strategy. *Evol. Syst.* **10**, 63–78 (2019).
23. Tharwat, A. & Schenck, W. Balancing exploration and exploitation: A novel active learner for imbalanced data. *Knowl.-Based Syst.* **210**, 106500 (2020).
24. Baum, E. B. & Lang, K. Query learning can work poorly when a human oracle is used. *Int. Jt Conf. Neural Netw.* **8**, 8 (1992).

25. Liu, W., Zhang, H., Ding, Z., Liu, Q. & Zhu, C. A comprehensive active learning method for multiclass imbalanced data streams with concept drift. *Knowl.-Based Syst.* **215**, 106778 (2021).
26. Zhang, B., Wang, Y. & Chen, F. Multilabel image classification via high-order label correlation driven active learning. *IEEE Trans. Image Process.* **23**, 1430–1441 (2014).
27. Ye, C. *et al.* Multi-label active learning with chi-square statistics for image classification. In *Proc. of the 5th ACM on International Conference on Multimedia Retrieval*, 583–586 (2015).
28. Rafiei, F. et al. Cfssynergy: Combining feature-based and similarity-based methods for drug synergy prediction. *J. Chem. Inf. Model.* **64**, 2577–2585 (2024).
29. Gharizadeh, A., Abbasi, K., Ghareyazi, A., Mofrad, M. R. & Rabiee, H. R. Hgtdr: Advancing drug repurposing with heterogeneous graph transformers. Preprint at http://arxiv.org/abs/2405.08031 (2024).
30. Esuli, A. & Sebastiani, F. Active learning strategies for multi-label text classification. In *European Conference on Information Retrieval*, 102–113 (Springer, 2009).
31. Singh, M., Brew, A., Greene, D. & Cunningham, P. *Score normalization and aggregation for active learning in multi-label classification* (University College Dublin, Tech. Rep, 2010).
32. Yang, B., Sun, J.-T., Wang, T. & Chen, Z. Effective multi-label active learning for text classification. In *Proc. of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 917–926 (2009).
33. Cherman, E. A., Tsoumakas, G. & Monard, M.-C. Active learning algorithms for multi-label data. In *IFIP International Conference on Artificial Intelligence Applications and Innovations*, 267–279 (Springer, 2016).
34. Ren, P. *et al.* A survey of deep active learning. Preprint at http://arxiv.org/abs/2009.00236 (2020).
35. Möllenbrok, L., Sumbul, G. & Demir, B. Deep active learning for multi-label classification of remote sensing images. *IEEE Geosci. Remote Sens. Lett.* (2023).
36. Wang, Q., Wu, W., Qi, Y. & Xin, Z. Combination of active learning and self-paced learning for deep answer selection with bayesian neural network. In *ECAI 2020*, 1587–1594 (IOS Press, 2020).
37. Saran, A., Yousefi, S., Krishnamurthy, A., Langford, J. & Ash, J. T. Streaming active learning with deep neural networks. In *International Conference on Machine Learning*, 30005–30021 (PMLR, 2023).
38. Yuan, D. *et al.* Active learning for deep visual tracking. *IEEE Trans. Neural Netw. Learn. Syst.* (2023).
39. Neal, R. M. *Bayesian learning for neural networks*, vol. 118 (Springer Science & Business Media, 2012).
40. Gal, Y. Uncertainty in deep learning. *University of Cambridge***1** (2016).
41. Zahera, H. M., Elgendy, I. A., Jalota, R. & Sherif, M. A. Fine-tuned bert model for multi-label tweets classification. In *TREC* (2019).
42. Yang, P. *et al.* Sgm: sequence generation model for multi-label classification. Preprint at http://arxiv.org/abs/1806.04822 (2018).
43. Yehuda, O., Dekel, A., Hacohen, G. & Weinshall, D. Active learning through a covering lens. *Adv. Neural. Inf. Process. Syst.* **35**, 22354–22367 (2022).

Acknowledgements

This paper is supported by National Natural Science Foundation of China (No. U22B2025, 62206279), Innovation Foundation for Doctor Dissertation of Northwestern Polytechnical University (No. CX2024019).

Author contributions

Q.W.: Conceptualization, Methodology, Validation, Writing-original draft. H.Z.: Methodology, Visualization, Writing-review & editing. W.Z.: Data curation, Funding acquisition. L.D.: Visualization, Validation. Y.L.: Formal analysis, Writing-review & editing. H.S.: Project administration, Supervision. Note that Qunbo Wang and Hangu Zhang contributed equally to this research.

Declarations

Competing interests

The authors declare no competing interests.

Additional information

Correspondence and requests for materials should be addressed to H.S.

Reprints and permissions information is available at www.nature.com/reprints.

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.