Fast Direct Search in an Optimally Compressed Continuous Target Space for Efficient Multi-Label Active Learning

Weishi Shi [1] **Qi Yu** [1]

Abstract

Active learning for multi-label classification poses fundamental challenges given the complex label correlations and a potentially large and sparse label space. We propose a novel CS-BPCA process that integrates compressed sensing and Bayesian principal component analysis to perform a two-level label transformation, resulting in an optimally compressed continuous target space. Besides leveraging correlation and sparsity of a large label space for effective compression, an optimal compressing rate and the relative importance of the resultant targets are automatically determined through Bayesian inference. Furthermore, the orthogonality of the transformed space completely decouples the correlations among targets, which significantly simplifies multi-label sampling in the target space. We define a novel sampling function that leverages a multi-output Gaussian Process (MOGP). Gradient-free optimization strategies are developed to achieve fast online hyper-parameter learning and model retraining for active learning. Experimental results over multiple real-world datasets and comparison with competitive multi-label active learning models demonstrate the effectiveness of the proposed framework.

1. Introduction

Multi-label classification (ML-C) aims to learn a model that automatically assigns *a set* of relevant labels to a data instance (Zhu et al., 2018; Liu et al., 2018; 2017). Multi-label problems naturally arise in many domains. For example, social media websites, including Twitter, Facebook, and Linked-in, assign tags to social media items, such as tweets, images, and user profiles, which can facilitate information retrieval and organization. Users from Q&A websites, such as stack overflow and Quora, are encouraged to choose tags from thousands of candidates to increase the exposure rate of their proposed questions. In bioinformatics, genes can be associated with multiple functional labels, such as metabolism and protein synthesis. Similarly, many image classification and video/audio recognition tasks are also multi-label problems.

A straightforward way to tackle the ML-C problems is to extend the single-label classification models by building binary relevance machines (BRMs) that construct an individual model for each label in a *one-versus-the-rest* manner (Tsoumakas et al., 2009). Significant effort has also been devoted by leveraging label correlations using techniques such as label propagation (Bi & Kwok, 2013) and transformation (Zhou et al., 2012). While ML-C remains as an active research area, a central component required by most ML-C models is a high-quality labeled dataset for model training. This falls under the broader task of *active learning*, where the idea is that by carefully choosing the most *informative* data instances for labelling, rather than in a purely random manner, better models can be trained with less labelling effort (Settles, 2012). There has been a wealth of work on active learning for single-label problems with state-of-the-art performance (Guo & Greiner, 2007; Ghani, 2002; Siddiquie & Gupta, 2010).

However, work on active learning for ML-C problems remains rather limited as it faces additional challenges when compared with single-label problems. In the latter, classes are assumed to be non-overlapping (i.e., only one label is assigned to each instance), so a data sample's overall contribution to all classes is essentially the sum of its contribution to each individual class, if being labeled. In multi-label problems, a good informativeness measure that quantifies a data sample's overall contribution to a correlated label space is much harder to design. Furthermore, the label space is usually highly sparse with many rare labels. Identifying data samples that help detect rare labels is much more challenging (due to lack of positive instances) but can provide special values (e.g., diagnosis of rare diseases). It is important to systematically leverage label correlations as they can provide information complementary to the scarce posi-

[1]Golisano College of Computing and Information Sciences, Rochester Institute of Technology, Rochester, USA. Correspondence to: Weishi Shi <ws7586@rit.edu>, Qi Yu <qi.yu@rit.edu>.

Proceedings of the 36th International Conference on Machine Learning, Long Beach, California, PMLR 97, 2019. Copyright 2019 by the author(s).

tive instances that contribute to the detection of rare labels. Finally, the computation cost of evaluating the informativeness measure may quickly become infeasible for real-time active learning tasks with the increase of the number of labels and/or the number of unlabeled data candidates.

In this paper, we develop a novel framework that simultaneously addresses all the key challenges as outlined above for multi-label active learning. First, a two-level label transformation is performed to generate a weighted, orthogonal, and continuous target space that significantly facilitates multi-label data sampling. This transformation is achieved by a coupled CS-BPCA process that integrates compressed sensing (CS) and Bayesian principal component analysis (BPCA). In this process, CS is responsible for converting a large, sparse, and correlated label space into a compact and continuous target space, where correlations among original labels are systematically leveraged for compressing purpose. BPCA further ensures the orthogonality of the resulting targets. Furthermore, the optimal size of the transformed target space and the relative importance of different targets can be obtained through Bayesian inference. By coupling CS with BPCA, the proposed process automatically infers a compressing rate optimal for active learning model training. In addition, the orthogonality of the transformed space completely de-correlates the resultant targets, which can significantly simplify multi-label sampling. We further define a novel sampling function that leverages a multi-output Gaussian Process (MOGP). By adopting a flexible covariance function, MOGP can capture the covariance structure of the input data precisely through continuous optimization of the hyper-parameters along with active learning. However, classical gradient ascent based approaches incur a prohibitive computational cost, which makes online sampling infeasible. We develop gradient-free hyper-parameter optimization by making novel extensions to two direct search methods including Bayesian optimization and simplex method.

In sum, our main contribution is threefold: (1) a CS-BPCA process that produces a compressed and orthogonal target space with optimal dimensionality to support multi-label data sampling, (2) a MOGP based sampling function that precisely captures the covariance structure of input data, and (3) gradient-free hyper-parameter optimization to enable fast online sampling for real-time active learning. The first two technical components are designed specifically to address the difficulty in designing good informative measures in multi-label active learning while the last component is to reduce the high-computational cost for a large label space. Extensive experiments are conducted over real-world multi-label datasets with distinct characteristics. Comparison with competitive multi-label active learning models helps demonstrate the effectiveness of the proposed framework, including overall active learning performance, sampling efficiency, and the ability to detect rare labels.

2. Related Work

This work is closely related to compressed sensing (CS), active learning, and Gaussian processes. Most relevant work fall into these categories is reviewed in this section.

In (Hsu et al., 2009), CS is adopted to solve ML-C problems. The proposed approach first projects the label vectors through CS. It then trains multiple regression models for a much smaller set of compressed labels. Among different label transformation approaches that reduce a large label space to a more compact one, CS appears to be the most efficient solution. The ability to perform fast label transformation makes CS an attractive component for active learning where label transformation needs to be conducted frequently.

Uncertainty sampling is commonly used to measure the informativeness of an unlabeled data instance for active learning (Cohn et al., 1996). For example, the predictive variance of Gaussian processes has been used for sampling in single-target regression (Krause & Guestrin, 2007) and multi-class classification (Kapoor et al., 2007). In multiple-target regression, the predictive entropy of a multi-output GP can be used as a sampling criterion for active learning. However, this conventional entropy criterion scales poorly with the number of labels. Some optimization and/or approximation strategies have been developed to improve the sampling performance (Zhang et al., 2016) but with a focus on regression tasks instead of ML-C.

Yang et.al. propose to use Maximum loss reduction with Maximal Confidence (MMC) as the sampling criterion for multi-label active learning (Yang et al., 2009). The loss reduction is evaluated as the sum of the expected classification error of individual SVMs and the true label of each candidate is estimated using logistic regression. However, MMC assumes independence among labels, which usually does not hold for most MC-L problems. Some recent works combine cross-class classification margin or aggregated uncertainty from BRMs with label inconsistency during multi-label sampling (Li & Guo, 2013; Reyes et al., 2018). Their improved performance implies that labels need to be considered jointly in multi-label settings. However, limited by the BRMs structure, their approach can not exploit the label correlation adequately. Multi-label active learning has also been considered in other tasks, including crowdsourcing (Li et al., 2015) and novel queries (Huang et al., 2015).

Some other approaches put ML-C problems in a fully Bayesian treatment, where the features and the labels are connected by finite latent variables. Then, the entropy can be inferred efficiently via variational inference (Kapoor et al., 2012; Vasisht et al., 2014). To ensure tractable inference, the potential functions assume an exponential form as being conjugate to the prior distributions of the latent variables. This essentially corresponds to a single RBF kernel in a

GP. The lack of choice for more flexible kernels prevents the model from more precisely capturing the covariance structure in the data.

3. Multi-label Active Learning

Let $X \in \mathbb{R}^{m \times n}$ be a training dataset with m data instances and n features and $Y \in \{0,1\}^{m \times l}$ be the labels where l is the total possible labels and $Y_{i,j} = 1$ indicates the i-th data instance is assigned label j. In a typical ML-C task (e.g., assign tags to images), the label matrix Y is usually very sparse: $\forall \boldsymbol{x}_i \in X : 1 \leq \sum_j Y_{i,j} \ll l$. Label sparsity and a potential large label space pose key challenges for training ML-C models, as explained above.

3.1. Weighted Orthogonal Label Space Transformation

To achieve fast and accurate sampling from a large and sparse label space, a key innovation of the proposed active learning framework is to generate a compact and continuous target space, where correlations among different targets are completely decoupled. While compressed sensing (CS) and related techniques have been leveraged for multi-label classification (Kapoor et al., 2007) and active learning tasks (Kapoor et al., 2012) with promising results, there are two key remaining challenges. First, there lacks a systematic way to determine an optimal compressing rate, which is typically obtained through cross-validation. However, active learning makes this more challenging as the optimal rate may be changing as the model is continuously updated. Second, the correlation may remain in the compressed target space, making sampling functions hard to design and/or expensive to implement. For example, a data sample contributes well to two highly correlated labels should be less preferred than the one that contributes well to two independent labels. Furthermore, not all the labels are equally important for the overall multi-label classification. The proposed CS-BPCA process addresses all these challenges by generating a *weighted orthogonal target space* for multi-label sampling, where an optimal compressing rate and relative importance of different targets are simultaneously achieved through Bayesian inference.

CS-BPCA performs a two-level transformation to arrive at the weighted orthogonal target space U: $Y \rightarrow R \rightarrow U$. In particular, CS converts the sparse and discrete label matrix Y to a dense and continuous matrix $R \in \mathbb{R}^{m \times d}$ where $d < l$. Since the transformation is linear, it can be represented by a matrix $A \in \mathbb{R}^{l \times d}$ and we have $R = YA$. This transformation not only removes the sparsity from the original label space Y, but also automatically encodes the correlations between labels into the compressed matrix R. We set the compressing rate $\frac{d}{l}$ to be relatively large (0.5 for our experiments), which will ensure high-quality recovery of original labels. The optimal compressing rate will be achieved by further compressing R using BPCA, resulting in $U \in \mathbb{R}^{m \times p}$, where $p < d$. Data sampling is directly conducted in this compressed target space with p independent targets that correspond to the p mutually orthogonal columns in U. Targets that aggregate information from important labels are assigned higher weights through BPCA, which will be encoded in the sampling function.

To make the proposed active learning framework practically useful, it should be able to assign labels in the original label space to new data instance $\boldsymbol{x}$. Since the prediction is in the target space U, two-level backward transformation will be performed: $\boldsymbol{u}_x \rightarrow \boldsymbol{r}_x \rightarrow \boldsymbol{y}_x$. First, $\boldsymbol{u}_x \rightarrow \boldsymbol{r}_x$ can be easily computed since BPCA essentially performs a linear and almost lossless projection. We then recover $\boldsymbol{y}_x$ from $\boldsymbol{r}_x$. It has been proved that a signal of length d in the compressed space can be efficiently recovered back to a k-sparse signal of length l by l_2 convex optimization (Candes & Tao, 2005) (additional details are provided in the supplemental materials). Figure 1 shows the overall CS-BPCA process. It highlights the roles of different components and how labels are transformed back and forth from the original label space to the target space for sampling and prediction.

To ensure a high-quality recovery (and accurate prediction) of the original labels, we set the compressing rate $\frac{d}{l}$ to a relatively high value and achieve an optimal compressing rate $\frac{p}{l}$ by compressing $\boldsymbol{r}$ output by CS using BPCA. In particular, assume that the distribution of the CS compressed output $\boldsymbol{r}$ is Gaussian of the form

$$p(\boldsymbol{r}|\boldsymbol{u}) = \mathcal{N}(\boldsymbol{r}|W\boldsymbol{u} + \boldsymbol{\mu}, \sigma^2 \boldsymbol{I}) \tag{1}$$

where $W \in \mathbb{R}^{d \times p} (p \leq d)$, $\boldsymbol{\mu} \in \mathbb{R}^{d \times 1}$, and σ is a scalar. $\boldsymbol{u} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I})$ is the latent representation of $\boldsymbol{r}$ in the space spanned by the columns of W. According to the linear Gaussian system, the marginal distribution of $\boldsymbol{r}$ is also a Gaussian, given by

$$p(\boldsymbol{r}) = \int p(\boldsymbol{r}|\boldsymbol{u})p(\boldsymbol{u})d\boldsymbol{u} = \mathcal{N}(\boldsymbol{r}|\boldsymbol{\mu}, WW^T + \sigma^2 \boldsymbol{I}) \tag{2}$$

The maximum likelihood (ML) solution of W is given by

$$W_{ML} = E_p(L_p - \sigma^2 \boldsymbol{I})^{\frac{1}{2}} O \tag{3}$$

where E_p consists of p eigenvectors of the covariance matrix of the compressed output R and the eigenvectors are those with the largeset eigenvalues, $\lambda_1 \geq ...\lambda_p$. $L_p = \text{diag}(\lambda_1, ...\lambda_p)$ and O is an arbitrary orthogonal matrix. The final compressed targets are obtained as the posterior mean:

$$\langle \boldsymbol{u} \rangle = M^{-1} W_{ML}^T (\boldsymbol{r} - \bar{\boldsymbol{r}}) \tag{4}$$

where $M = W^T W$ and the expectation $\langle \cdot \rangle$ is computed over $p(\boldsymbol{u}|\boldsymbol{r})$, which is also a Gaussian. (4) reduces to the result of a conventional PCA as $\sigma^2 \rightarrow 0$, which implies an orthogonal projection of R into the final target space

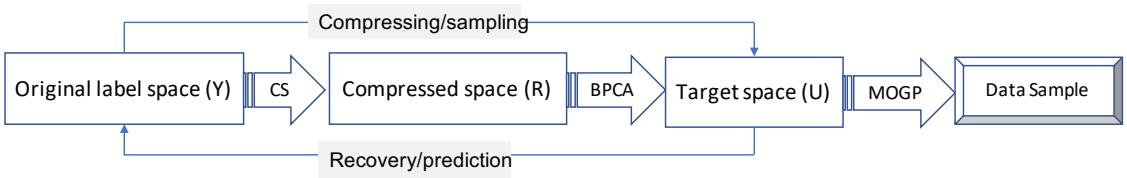

Figure 1. The two-level label transformation by the CS-BPCA process

U. The significance of the i-th target is weighted by the corresponding eigenvalue λ_i, which is guaranteed to be non-negative since the covariance matrix is positive semidefinite (PSD). For $\sigma^2 > 0$, orthogonality still holds except that the projection is shifted towards the origin (Bishop, 2006).

Instead of manually setting the dimensionality p of targets $\boldsymbol{u}$, BPCA places a prior over W, given by

$$p(W|\boldsymbol{\alpha}) = \prod_{i=1}^{p} (\frac{\alpha_i}{2\pi})^{\frac{d}{2}} e^{-\frac{1}{2}\alpha_i ||\boldsymbol{w}_i||^2} \tag{5}$$

where p is initialized to its maximum possible value $p = d - 1$. The redundant dimension $\boldsymbol{w}_i$ will be pruned automatically as its corresponding precision distribution α_i converges to a large value after the optimization. By adding priors over $\boldsymbol{\mu}$, σ^{-2}, and $\boldsymbol{\alpha}$, we achieve a complete Bayesian model: $p(\boldsymbol{\mu}) = \mathcal{N}(\mathbf{0}, \beta^{-1}\boldsymbol{I}), p(\boldsymbol{\alpha}) = \prod_{i=1}^{p} \Gamma(\alpha_i | a_{\boldsymbol{\alpha}}, b_{\boldsymbol{\alpha}}), p(\sigma^{-2}) = \Gamma(\sigma^{-2}|c_\sigma, d_\sigma)$. Inference can be performed through variational inference by using a factorized variational distribution: $Q(U, W, \boldsymbol{\mu}, \sigma^{-2}, \boldsymbol{\alpha}) = Q(U)Q(W)Q(\boldsymbol{\mu})Q(\sigma^{-2})Q(\boldsymbol{\alpha})$.

3.2. Data Sampling in the Transformed Target Space

As the transformed space consists of p targets, we choose a multi-output GP (MOGP) for data sampling for two major reasons. First, the overall informativeness (or uncertainty) of a data sample can be quantified through the covariance of the multi-output predictive distribution. Second, by adopting a flexible covariance function, MOGP can capture the covariance structure of the input data precisely by continuously optimizing the hyper-parameters along with active learning. We further develop gradient-free methods (see next section for details) for hyper-parameter optimization that enables fast online sampling in active learning.

A general MOGP places a GP prior over a set of latent functions $\{f^{(1)}, ..., f^{(p)}\}$ (Bonilla et al., 2007). It is typical to assume zero mean and we have

$$\langle f^{(g)}(\boldsymbol{x}) f^{(h)}(\boldsymbol{x}') \rangle = K^f_{g,h} k^x(\boldsymbol{x}, \boldsymbol{x}') \tag{6}$$

$$\boldsymbol{u}_i^{(g)} \sim \mathcal{N}(f^{(g)}(\boldsymbol{x}_i), \beta_g^{-1}) \tag{7}$$

where K^f is a PSD matrix with $K^f_{g,h}$ capturing the correlation between targets $\boldsymbol{u}^{(g)}$ and $\boldsymbol{u}^{(h)}$, k^x is a covariance function over inputs, and β_g is the precision of the g-th target. The predictive distribution over a new data point $\boldsymbol{z}$ is also a Gaussian with mean and covariance given by

$$\boldsymbol{m}(\boldsymbol{z}) = (K^f \otimes \boldsymbol{k}_z^x)^T C^{-1} \operatorname{vec}(U) \tag{8}$$

$$C(\boldsymbol{z})_{g,h} = K^f_{g,h} k^x(\boldsymbol{z}, \boldsymbol{z}) + D_{g,h} - (\boldsymbol{k}_g^f \otimes \boldsymbol{k}_z^x)^T C^{-1} (\boldsymbol{k}_h^f \otimes \boldsymbol{k}_z^x) \tag{9}$$

where $\otimes$ is the Kronecker product, $\boldsymbol{k}_z^x = (k^x(\boldsymbol{x}_1, \boldsymbol{z}), ..., k^x(\boldsymbol{x}_m, \boldsymbol{z}))^T$, $\boldsymbol{k}_g^f$ is the g-th column of K^f, $D = \operatorname{diag}(\beta_1, ..., \beta_p)^T$, and $C = K^f \otimes K^x + D \otimes I$ with K^x being the covariance matrix of the m training instances.

Intuitively, the most informative data sample should be the one with the maximum uncertainty for the active learner. As a result of labelling such a sample, the active leaner can be improved the most. Since the predictive distribution of the MOGP jointly considers all the targets, data sampling in the transformed target space can be achieved using the predictive entropy of the MOGP:

$$\boldsymbol{z}^* = \arg\max_{\boldsymbol{z} \in X_u} H(\boldsymbol{z}) = \arg\max_{\boldsymbol{z} \in X_u} \ln(|C(\boldsymbol{z})|) \tag{10}$$

where X_u denotes a pool of unlabeled data samples. A fundamental challenge of data sampling using a general MOGP is the prohibitive computational cost for evaluating (10). A central part of the computation involves the inverse of a $pm \times pm$ covariance matrix C given by (9). Furthermore, both the target correlation matrix K^f and the hyper-parameters of the covariance function k^x need to be learned by optimizing the marginal likelihood of the targets. Such optimization is typically performed through an iterative gradient based approach, which also requires to compute C^{-1} in each iteration. The high computational cost prevents using the above sampling criteria for active learning, where parameter learning and model training are conducted on the fly as new data samples are continuously being labelled.

Fortunately, using a MOGP built from the transformed target space can significantly reduce the computational cost given that different targets are mutually orthogonal through BPCA projection. As a result, the covariance matrix C has a block structure, which allows us to treat each target independently. More specifically, K^f reduces to an identity matrix and the predictive distribution of the g-th task is defined by its mean and covariance, given by

$$m^{(g)}(\boldsymbol{z}) = (\boldsymbol{k}_z^x)^T (C^{(g)})^{-1} \boldsymbol{u}^{(g)} \tag{11}$$

$$C(\boldsymbol{z})_{g,g} = \boldsymbol{k}^x(\boldsymbol{z}, \boldsymbol{z}) + \beta_g^{-1} - (\boldsymbol{k}_z^x)^T (C^{(g)})^{-1} \boldsymbol{k}_z^x \tag{12}$$

where $C^{(g)}$ is the $m \times m$ covariance matrix of the g-th target and $(C^{(g)})^{-1}$ is much cheaper to evaluate than C^{-1}. We will develop efficient algorithms for learning hyper-parameters of the kernel functions in next section. By further integrating the weights of different targets, we arrive at the following sampling function in the transformed target space:

$$\boldsymbol{z}^* = \arg\max_{\boldsymbol{z} \in X_u} \left(-\sum_{g=1}^{p} \lambda_g \ln C(\boldsymbol{z})_{g,g} \right) \tag{13}$$

3.3. Gradient-free Hyper-parameter Optimization

We choose the following kernel with four hyper-parameters $\boldsymbol{\theta} = (\theta_0, \theta_1, \theta_2, \theta_3)$:

$$k(\boldsymbol{x}_i, \boldsymbol{x}_j) = \theta_0 \exp\{-\frac{\theta_1}{2} \|\boldsymbol{x}_i - \boldsymbol{x}_j\|^2\} + \theta_2 \boldsymbol{x}_i^T \boldsymbol{x}_j + \theta_3 \tag{14}$$

It should be noted that a different $\boldsymbol{\theta}$ will be learned for a different target. Since the learning algorithm is the same, we remove the superscript to make the notation uncluttered. The non-kernel based regression models require at least np parameters to determine p different regressors and the effect of over-fitting will be amplified by p times when new basis functions are introduced to increase the flexibility of those models. On the other hand, with the customized kernel, there are only $4p$ parameters that need to be learned.

Gradient based approaches are commonly used to learn the hyper-parameters. In particular, the log likelihood of a GP with kernel (14) is given by

$$\mathcal{L}(\boldsymbol{\theta}) = \ln p(\mathbf{u}|\boldsymbol{\theta}) = -\frac{1}{2} \ln |C| - \frac{1}{2} \mathbf{u}^T C^{-1} \mathbf{u} + \text{const.} \tag{15}$$

Taking the partial derivative of the log likelihood, we have

$$\frac{\partial}{\partial \theta_i} \ln p(\mathbf{u}|\boldsymbol{\theta}) = -\frac{1}{2} \operatorname{Tr}\left(C^{-1} \frac{\partial C}{\partial \theta_i} \right) + \frac{1}{2} \mathbf{u}^T C^{-1} \frac{\partial C}{\partial \theta_i} C^{-1} \mathbf{u} \tag{16}$$

The time complexity of a gradient ascent method is $O(|\boldsymbol{\theta}| m^3)$ as it involves evaluation of C^{-1}. Since $\mathcal{L}(\boldsymbol{\theta})$ is non-convex, gradient ascent usually needs to run multiple times with different random initialization to avoid a local optimal with poor quality. Furthermore, for high-dimensional data (e.g., $n > m$), since Σ needs to be reconstructed for a new $\boldsymbol{\theta}$, the construction cost should also be considered, leading to an overall complexity of $O(|\boldsymbol{\theta}|(m^3 + m^2 n))$.

While gradient based approaches can be used to train a regular GP for classification/regression, they are no longer suitable for active learning where the hyper-parameters need to be learned on the fly as the model is being continuously updated as more data samples are being labelled. In fact, re-learning the hyper-parameters is essential for a GP active learner so that it can precisely capture the covariance structure of currently labeled data for accurate data sampling. We develop novel gradient-free optimization strategies to significantly reduce the computational cost for hyper-parameter learning. In particular, we leverage two direct search optimization approaches, Bayesian optimization and simplex methods, and make key extensions to achieve fast sampling.

Bayesian Optimization (B-OPT) B-OPT aims to select a $\boldsymbol{\theta}^*$ from a grid search space that maximizes $\mathcal{L}(\boldsymbol{\theta})$. Since $\mathcal{L}(\boldsymbol{\theta})$ is expensive to compute, B-OPT trains a probabilistic model $\mathcal{M}$ and uses its predictive distribution $p(\mathcal{L}(\boldsymbol{\theta})|\boldsymbol{\theta}) \sim \mathcal{N}(m_{\boldsymbol{\theta}}, \sigma^2_{\boldsymbol{\theta}})$ to estimate $\mathcal{L}(\boldsymbol{\theta})$. An acquisition function is used to measure whether the predicted log-likelihood value of $\boldsymbol{\theta}$ exceeds some threshold $\mathcal{L}^*(\boldsymbol{\theta})$: $f(\mathcal{L}(\boldsymbol{\theta})) = \max((\mathcal{L}(\boldsymbol{\theta}) - \mathcal{L}^*(\boldsymbol{\theta})), 0)$ Then, the expected improvement (Jones, 2001) is used as a cheap surrogate of $\mathcal{L}(\boldsymbol{\theta})$ to choose a candidate $\boldsymbol{\theta}$ from the grid search space.

$$EI(\boldsymbol{\theta}) = \int_{-\infty}^{\infty} f(\mathcal{L}(\boldsymbol{\theta})) p(\mathcal{L}(\boldsymbol{\theta})|\boldsymbol{\theta}) d\mathcal{L}(\boldsymbol{\theta}) \tag{17}$$

Locatelli has proved that the iterates from above sampling method is guaranteed to converge to a global optimal (Locatelli, 1997), which makes the direct search of $\boldsymbol{\theta}^*$ an active learning-like process. At each searching iteration, $\boldsymbol{\theta}^*$ is first selected by (17). Then, its true log-likelihood will be evaluated to update $\mathcal{M}$ in (17) for next iteration. In our approach, we choose the threshold $\mathcal{L}^*(\boldsymbol{\theta})$ to be the maximum log-likelihood value of the current observations and use 95% confidence interval to compute (17):

$$EI_{95\%}(\boldsymbol{\theta}) = \max(\mathcal{L}(m_{\boldsymbol{\theta}} \pm 1.96\sigma_{\boldsymbol{\theta}}) - \mathcal{L}^*(\boldsymbol{\theta}), 0) \tag{18}$$

Simplex Optimization (S-OPT) One drawback of B-OPT is that the volume of the grid search space grows exponentially as we attempt to expand the searching range or refine the searching granularity. Such a large space is expensive to be stored and searched. The simplex method (Bertsekas, 1999) overcomes such a drawback as it does not require to build the search space explicitly. The method starts with a simplex, which is the convex combination of $|\boldsymbol{\theta}| + 1$ initial points in the search space. The worst and best vertices of the simplex satisfy the following conditions:

$$\boldsymbol{\theta}_{min} = \arg\min_{i=0,..,|\boldsymbol{\theta}|} \mathcal{L}(\boldsymbol{\theta}_i) \quad \boldsymbol{\theta}_{max} = \arg\max_{i=0,..,|\boldsymbol{\theta}|} \mathcal{L}(\boldsymbol{\theta}_i) \tag{19}$$

Let $\hat{\boldsymbol{\theta}}$ denote the centroid of the simplex formed by all vertices but $\boldsymbol{\theta}_{min}$: $\hat{\boldsymbol{\theta}} = \frac{1}{n}(\sum_{i=0}^{n} \boldsymbol{\theta}_i - \boldsymbol{\theta}_{min})$. The simplex method works by iteratively replacing $\boldsymbol{\theta}_{min}$ using a new $\boldsymbol{\theta}_{new}$, so that $\mathcal{L}(\boldsymbol{\theta}_{new}) > \mathcal{L}(\boldsymbol{\theta}_{min})$. The search of $\boldsymbol{\theta}_{new}$ is performed through expansion, reflection, or contraction of the simplex so that it will be moved in the direction where the objective function $\mathcal{L}$ can be improved. The detailed process for searching $\boldsymbol{\theta}_{new}$ is given in the supplemental materials.

The major cost in B-OPT and S-OPT lies in computing $\mathcal{L}(\boldsymbol{\theta})$, which involves evaluating the determinant and inverse of the covariance matrix Σ with a time complexity of $O(m^3)$. Their systematic search strategies ensure a fast convergence, which is evidenced by our experiments. Standard linear algebra techniques such as Cholesky decomposition and Nyström approximation can be applied to further speed up the computation. However, for high-dimensional active learning tasks (i.e., $n \gg m$), reconstructing the covariance matrix will dominate the computational cost. For each new candidate $\boldsymbol{\theta}$, Σ needs to be reconstructed by computing each entry according to (14) with a cost of $O(m^2 n)$. We develop a fast covariance matrix reconstruction approach to further reduce the computational cost.

Specifically, we separate two blocks of computation that are invariant to $\boldsymbol{\theta}$ from (14) and denote them as A and B, where $A_{i,j} = ||\boldsymbol{x}_i - \boldsymbol{x}_j||^2$ and $B_{i,j} = \boldsymbol{x}_i^T \boldsymbol{x}_j$. Plugging A and B in (14), we have $\Sigma = \theta_0 \exp\{-\frac{\theta_1}{2} A\} + \theta_2 B + \theta_3$. Since A and B are fixed and only $\boldsymbol{\theta}$ is updated, the reconstruction cost is reduced to m^2, which can be efficiently computed as m is typically small for active learning. Furthermore, when a newly labeled data sample $\boldsymbol{x}_{m+1}$ is included by active learning, both A and B can be efficiently updated instead of being recomputed from scratch:

$$A_{m+1} = \begin{pmatrix} A_m & \boldsymbol{a}_{m+1} \\ \boldsymbol{a}_{m+1}^T & 0 \end{pmatrix}, B_{m+1} = \begin{pmatrix} B_m & \boldsymbol{b}_{m+1} \\ \boldsymbol{b}_{m+1}^T & \boldsymbol{x}_{m+1}^T \boldsymbol{x}_{m+1} \end{pmatrix}$$

where $\boldsymbol{a}_{m+1}$ is a vector of squared distance between $\boldsymbol{x}_{m+1}$ and $\boldsymbol{x}_i$'s and $\boldsymbol{b}_{m+1}$ is vector of dot products.

4. Experiments

We conduct extensive experiments over five real-world multi-label datasets, aiming to: (i) explore the behaviours of our model under different sets of model parameters, (ii) demonstrate that our proposed model is superior to other state-of-the-art competitive multi-label active learning methods, and (iii) show the efficiency of the proposed gradient-free optimization strategies for active learning.

4.1. Datasets and Experimental Settings

We choose five representative real-world datasets from different domains. All datasets have a large and very sparse label space with a relatively high label cardinality, allowing us to properly evaluate multi-label active learning. Each dataset is partitioned into three parts: training, candidate pool, and testing. To ensure at least one positive data instance in each partition for proper model training/testing, we pre-process the datasets by first removing labels with less than 0.5% instances in the whole dataset. We then remove data instances with no positive labels after the previous step. Table 1 summarizes the key properties of the pre-processed datasets. We use 1% of the data to start active learning, 40% as unlabeled candidate pool, and the remaining for testing. The data is shuffled before splitting. We use Macro F-score (averaged F_1 score over all the labels) to evaluate the model performance and an average over three runs is reported.

4.2. Impact of Model Parameters

We plot the active learning performance of the proposed model under different compressing rate, kernel update period (KUP), which is the number of active learning iterations between two kernel optimization, and the sparsity level of the recovered labels, to investigate their impacts. Macro F-score is measured over the test data after the current model was updated with a newly labelled data instance.

Figure 2 shows the performance of the proposed model with different KUPs. In general, the model performance decreases with a larger KUP. This clearly demonstrates that the MOGP based sampling can capture the covariance structure of the input data precisely by continuously optimizing the hyper-parameters along with active learning. The best performance is obtained by the two proposed kernel optimization methods, B-OPT and S-OPT, which allow optimizing the GP kernels in each iteration (see the reported CPU times in a later section for details). In contrast, the gradient ascent method only affords to optimize the kernel much less frequently to make it a practical sampling approach for active learning.

Figure 3 shows the effectiveness of sampling in the compressed target space. We set the first level compressing rate as 0.5 and the final compressing rate is determined by BPCA. As this rate is dynamically adjusted with active learning, we report the average rate over all iterations. The other two curves are generated by only applying CS with a fixed compressing rate. In most cases, sampling in the compressed target space achieves a much better performance with a lower (and auto-determined) compressing rate. Finally, Figure 4 shows that the model performs the best when using the average label cardinality of the dataset as recovered label sparsity.

4.3. Performance Comparison

We compare with some competitive multi-label active learning models to demonstrate the effectiveness of the proposed framework (noted as CS-BPCA-GP in the figures). We include two types of active learning models that perform data sampling in either a compressed label space (Type I) or the original label space (Type II).

- **Type I models** generate a compressed label space (e.g., through CS) and then perform data sampling in the compressed space. We consider three models: Mutual information based sampling (CS-MIML) (Vasisht et al., 2014), Bayesian Ridge regression (CS-BR) over a compressed label space, and Ridge regression based random sampling (CS-RR) over a compressed label space.
- **Type II models** directly perform data sampling in the original label space. State of the art sampling perfor-

Table 1. Summary of Datasets

Dataset	Domain	Instances	Features	Labels	Min Label Support	Label Card.	Label Sparsity
Delicious	web	8172	500 (nominal)	157	208	5.56	0.03
BookMark	publication	38548	2150 (nominal)	136	442	3.45	0.02
WebAPI	software	9166	5659 (numeric)	90	71	2.50	0.02
Corel5K	images	5000	499 (nominal)	132	25	3.25	0.02
Bibtex	text	7013	1836 (nominal)	127	45	2.4	0.02

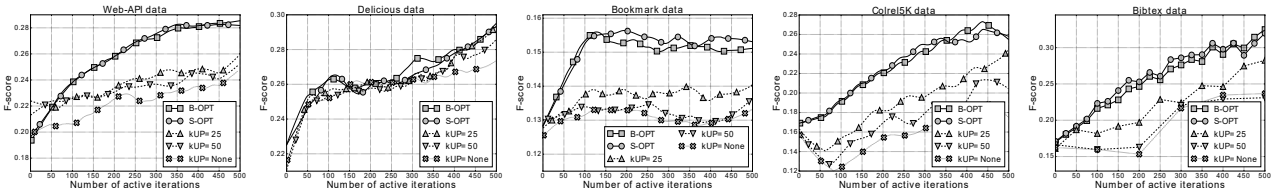

Figure 2. Impact of Kernel Optimization Frequency

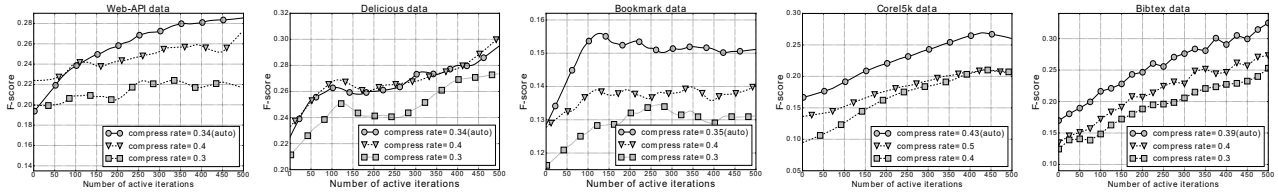

Figure 3. Impact of Compressing Rate

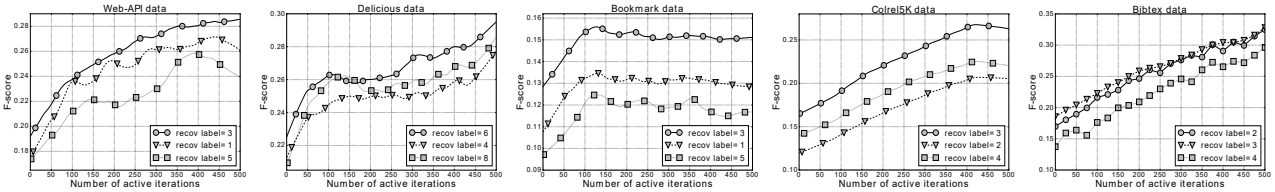

Figure 4. Impact of Recovered Label Sparsity

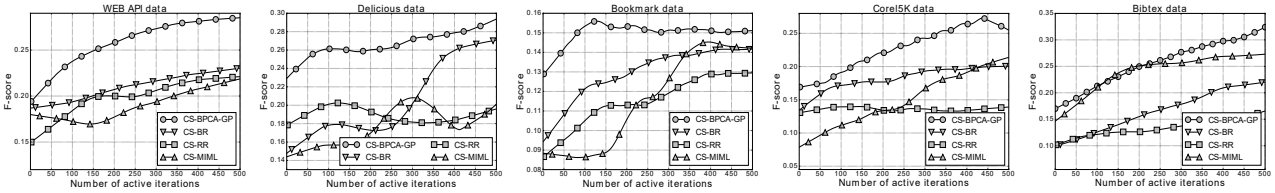

Figure 5. Comparison Result I

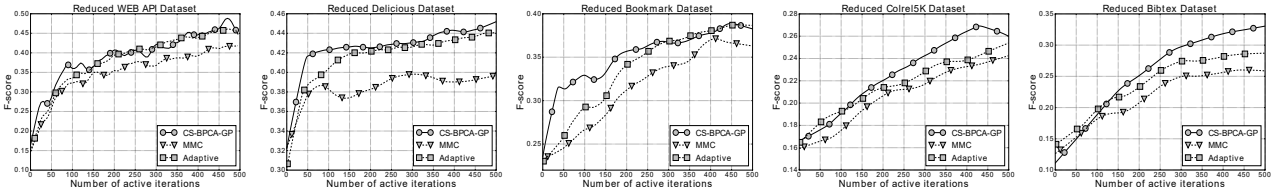

Figure 6. Comparison Result II

mance is achieved by BRMs based models, including MMC (Yang et al., 2009) and Adaptive (Li & Guo, 2013). However, the high computational cost may limit the applicability of these models to a large label space.

Figure 5 shows the comparison with Type I models, where all the labels in the data are used. The clear advantage of our approach is due to the combined contribution of optimal label space compression and the effective sampling criterion defined over the MOGP with optimized kernels. Informative data samples that provide maximum contribution to the learning of the entire label space can be effectively identified and labelled. As a result, it outperforms other compressing methods (CS only), none-GP (BR and RR), and GP with restrictive kernels (CS-MIML)

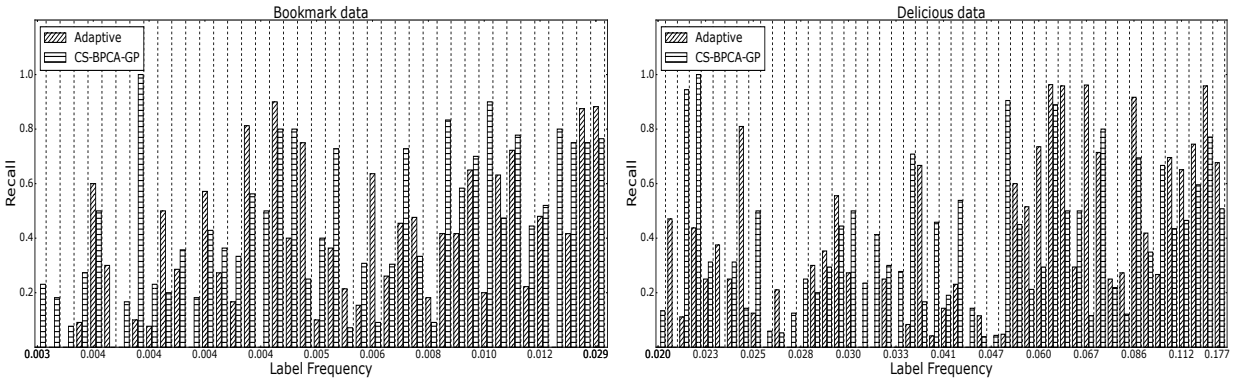

Figure 7. Rare Label Prediction Comparison

The BRMs based Type II models build a separate classifier for each label. Hence, the computational resource is quickly depleted when a large number of labels are involved. To allow data sampling to be done in a reasonable amount of time for practical active learning, we limit the number of labels for each dataset to be 50 by further removing some rare labels. This actually benefits the BRMs models as they tend to perform poorly over the rare labels without leveraging the label correlations. Figure 6 shows that CS-BPCA-GP outperforms the other two methods especially at the early stage of active learning.

It is also worth to note that both MMC and Adaptive perform well on relatively frequent labels, which consist of sufficient data instances to train an accurate model. However, they perform rather poorly on some rare labels that are much more difficult to predict. Figure 7 compares the recall performance between the proposed approach with the Adaptive model (the best one in Type II models) using Bookmark and Delicious data as examples (see the supplemental materials for more results). The Adaptive model completely fails to predict 7 and 9 labels (versus 1 and 2 for CS-BPCA-GP), respectively. This clearly demonstrates the advantage of CS-BPCA-GP for training ML-C models that better recover rare labels. Such models are more desirable as frequent labels are usually much easier to predict due to the ample positive training samples. Besides higher model accuracy, CS-BPCA-GP also shows a clear computational advantage through its optimal label transformation/compression. For all three datasets, the two Type II models take more than 10 hours while CS-BPCA-GP uses less than 2 hours to finish 500 active learning iterations.

4.4. Efficiency of Gradient-free Optimization

To demonstrate the effectiveness of the proposed gradient-free hyper-parameter optimization strategies, we make a comparison with the classical gradient ascent (GA) based method. Since the size of the kernel matrix changes as newly labeled data instances are added, we compute the average CPU time for hyper-parameter optimization and the results are given in Table 2. As can be seen, for the two larger datasets with more features and instances, GA spends more than 10s for optimizing the hyper-parameters. This, when coupled with other overhead (e.g., evaluating the sampling function (13)) may make it too slow for practical active learning. In contrast, the two gradient-free methods use less than 1s, which justifies their potential to support real-world large-scale active learning problems.

Table 2. CPU Time (s) of Hyper-parameter Optimization

Dataset	GA	B-OPT	S-OPT
Delicious	1.83	0.17	0.20
BookMark	15.0	0.80	0.79
WebAPI	10.10	0.54	0.55
Corel5K	0.58	0.08	0.08
Bibtex	8.71	0.48	0.51

5. Conclusion and Future Work

In this paper, we conduct novel label transformation that enables multi-label active learning to be performed in an optimally compressed target space. The mutually orthogonal targets significantly simplify evaluating the predictive entropy of a MOGP, which is used as the sampling criterion for choosing the most informative data instances over all the labels. Gradient-free optimization is developed for fast learning of hyper-parameters, which ensures the MOGP covariance to closely follow the frequently updated data for accurate data sampling. Extensive experiments conducted on real-world multi-label data demonstrate the effectiveness of the proposed framework. We identify two interesting future directions. First, we plan to achieve active diagnosis that will allow the annotation task to focus on the most informative label candidates of the selected data instance. This is extremely helpful when the number of candidate labels is huge. The model should also be able to utilize partially labelled data from active diagnosis for training purpose. Second, we suggest to infer the label recovery sparsity for individual data instances rather than using a fixed sparsity level. This will improve the model performance especially when the dataset has a large variance of label sparsity.

Acknowledgement

This research was supported in part by an NSF IIS award IIS-1814450 and an ONR award N00014-18-1-2875. The views and conclusions contained in this paper are those of the authors and should not be interpreted as representing any funding agency.

References

Bertsekas, D. P. *Nonlinear programming*. Athena scientific Belmont, 1999.

Bi, W. and Kwok, J. Efficient multi-label classification with many labels. In *International Conference on Machine Learning*, pp. 405–413, 2013.

Bishop, C. M. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2006. ISBN 0387310738.

Bonilla, E. V., Chai, K. M. A., and Williams, C. K. I. Multi-task gaussian process prediction. In *Proceedings of the 20th International Conference on Neural Information Processing Systems*, NIPS'07, pp. 153–160, 2007.

Candes, E. J. and Tao, T. Decoding by linear programming. *IEEE transactions on information theory*, 51(12):4203–4215, 2005.

Cohn, D. A., Ghahramani, Z., and Jordan, M. I. Active learning with statistical models. *Journal of artificial intelligence research*, 1996.

Ghani, R. Combining labeled and unlabeled data for multi-class text categorization. In *ICML*, volume 2, pp. 8–12, 2002.

Guo, Y. and Greiner, R. Optimistic active-learning using mutual information. In *IJCAI*, volume 7, pp. 823–829, 2007.

Hsu, D. J., Kakade, S. M., Langford, J., and Zhang, T. Multi-label prediction via compressed sensing. In *Advances in neural information processing systems*, pp. 772–780, 2009.

Huang, S.-J., Chen, S., and Zhou, Z.-H. Multi-label active learning: Query type matters. In *IJCAI*, pp. 946–952, 2015.

Jones, D. R. A taxonomy of global optimization methods based on response surfaces. *Journal of global optimization*, 21(4):345–383, 2001.

Kapoor, A., Grauman, K., Urtasun, R., and Darrell, T. Active learning with gaussian processes for object categorization. In *Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on*, pp. 1–8. IEEE, 2007.

Kapoor, A., Jain, P., and Viswanathan, R. Multilabel classification using bayesian compressed sensing. In *Proceedings of the 25th International Conference on Neural Information Processing Systems - Volume 2*, NIPS'12, pp. 2645–2653, 2012.

Krause, A. and Guestrin, C. Nonmyopic active learning of gaussian processes: an exploration-exploitation approach. In *Proceedings of the 24th international conference on Machine learning*, pp. 449–456. ACM, 2007.

Li, S.-Y., Jiang, Y., and Zhou, Z.-H. Multi-label active learning from crowds. *arXiv preprint arXiv:1508.00722*, 2015.

Li, X. and Guo, Y. Active learning with multi-label svm classification. In *IJCAI*, pp. 1479–1485, 2013.

Liu, C., Zhao, P., Huang, S.-J., Jiang, Y., and Zhou, Z.-H. Dual set multi-label learning. In *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

Liu, W., Tsang, I. W., and Müller, K.-R. An easy-to-hard learning paradigm for multiple classes and multiple labels. *The Journal of Machine Learning Research*, 18(1):3300–3337, 2017.

Locatelli, M. Bayesian algorithms for one-dimensional global optimization. *Journal of Global Optimization*, 10 (1):57–76, 1997.

Reyes, O., Morell, C., and Ventura, S. Effective active learning strategy for multi-label learning. *Neurocomputing*, 273:494–508, 2018.

Rudelson, M. and Vershynin, R. Sparse reconstruction by convex relaxation: Fourier and gaussian measurements. In *Information Sciences and Systems, 2006 40th Annual Conference on*, pp. 207–212. IEEE, 2006.

Settles, B. Active learning. *Synthesis Lectures on Artificial Intelligence and Machine Learning*, 6(1):1–114, 2012.

Siddiquie, B. and Gupta, A. Beyond active noun tagging: Modeling contextual interactions for multi-class active learning. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pp. 2979–2986. IEEE, 2010.

Tsoumakas, G., Katakis, I., and Vlahavas, I. Mining multi-label data. In *Data mining and knowledge discovery handbook*, pp. 667–685. Springer, 2009.

Vasisht, D., Damianou, A., Varma, M., and Kapoor, A. Active learning for sparse bayesian multilabel classification. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 472–481. ACM, 2014.

Yang, B., Sun, J.-T., Wang, T., and Chen, Z. Effective multi-label active learning for text classification. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 917–926. ACM, 2009.

Zhang, Y., Hoang, T. N., Low, K. H., and Kankanhalli, M. S. Near-optimal active learning of multi-output gaussian processes. In *AAAI*, pp. 2351–2357, 2016.

Zhou, T., Tao, D., and Wu, X. Compressed labeling on distilled labelsets for multi-label learning. *Machine Learning*, 88(1-2):69–126, 2012.

Zhu, Y., Kwok, J. T., and Zhou, Z.-H. Multi-label learning with global and local label correlation. *IEEE Transactions on Knowledge and Data Engineering*, 30(6):1081–1094, 2018.