Knowledge Amalgamation for Multi-Label Classification via Label Dependency Transfer

Jidapa Thadajarassiri[1], Thomas Hartvigsen[2], Walter Gerych[1], Xiangnan Kong[1], Elke Rundensteiner[1]

[1]Worcester Polytechnic Institute,
[2]Massachusetts Institute of Technology
{jthadajarassiri, wgerych, xkong, rundenst}@wpi.edu, tomh@mit.edu

Abstract

Multi-label classification (MLC), which assigns multiple labels to each instance, is crucial to domains from computer vision to text mining. Conventional methods for MLC require huge amounts of labeled data to capture complex dependencies between labels. However, such labeled datasets are expensive, or even impossible, to acquire. Worse yet, these pre-trained MLC models can only be used for the particular label set covered in the training data. Despite this severe limitation, few methods exist for expanding the set of labels predicted by pre-trained models. Instead, we acquire vast amounts of new labeled data and retrain a new model from scratch. Here, we propose combining the knowledge from multiple pre-trained models (*teachers*) to train a new *student* model that covers the union of the labels predicted by this set of teachers. This student supports a broader label set than any one of its teachers without using labeled data. We call this new problem knowledge amalgamation for multi-label classification. Our new method, **A**daptive K**N**owledge **T**ransfer (ANT), trains a student by learning from each teacher's partial knowledge of label dependencies to infer the global dependencies between all labels across the teachers. We show that ANT succeeds in unifying label dependencies among teachers, outperforming five state-of-the-art methods on eight real-world datasets.

Introduction

Multi-label classification (MLC) is crucial for real-world applications where instances are associated with multiple labels simultaneously. Examples of these applications include computer vision (Chen et al. 2019), text mining (Yang et al. 2018), and bioinformatics (Vens et al. 2008). The MLC task requires sophisticated solutions, as any approach must overcome the hurdle caused by many possible label subsets—which are exponential in the number of labels—that could be applied to any instance (Dembszynski et al. 2010).

State-of-the-Art. Modern multi-label classifiers perform remarkably well on this challenging task by exploiting dependencies between labels (Wang et al. 2016; Nam et al. 2017; Chen et al. 2018). To achieve this, recent methods build on Classifier Chains (Dembszynski et al. 2010; Read et al. 2011), which predict labels sequentially, conditioning the prediction of each label on all previously predicted labels. These approaches (Chen et al. 2018; Nam et al. 2019) use large RNNs to share parameters across label predictions.

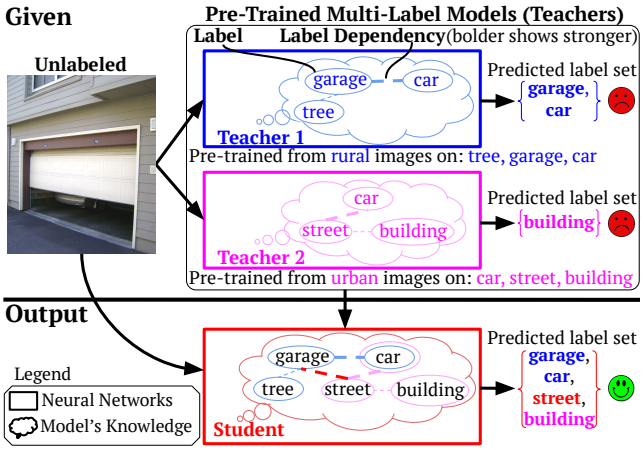
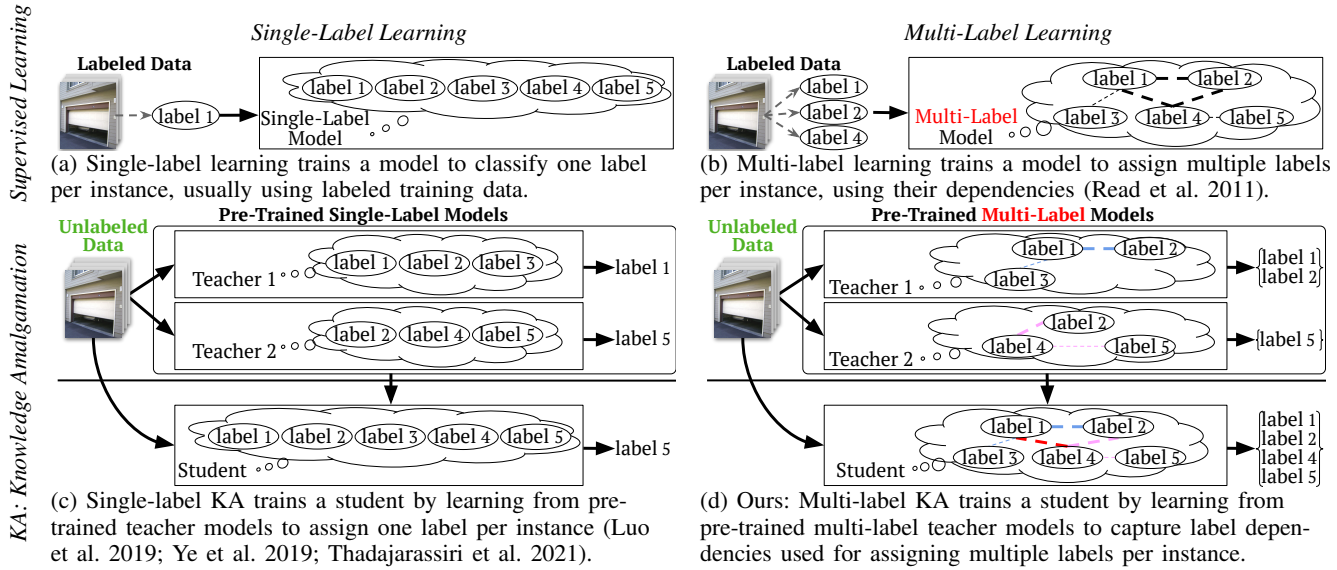


Figure 1: Knowledge amalgamation for multi-label classification. Given pre-trained multi-label models (*teachers*) and unlabeled data, the task is to train a *student* that accurately classifies labels in the union of teachers' specialized labels.

However, these works (Zhang and Zhang 2010; Tsai and Lee 2020) have only been developed for standard supervised learning, requiring a massive amount of labeled data to sufficiently learn label dependencies. Despite the expensive cost to gather such labeled data, the usability of the resulting model is limited to only the specific label set that it was pre-trained for and cannot adapt for use with broader label sets. Instead, the model would need to be retrained on a new labeled dataset, requiring practitioners to re-annotate all instances to consider also new labels or to acquire many more labeled instances to cover the new possible label sets.

To alleviate these costs, Knowledge Amalgamation (KA) (Ye et al. 2019) is a learning paradigm that, using only *unlabeled data*, combines the knowledge of multiple pre-trained models (*teachers*) into one *student*. The student then handles a broader task set than that of any of its teachers by covering the union of their labels. Ideally, KA could be used to combine the knowledge of *multi-label* classifiers to extend their knowledge-base without the expense of collecting more labeled data. This setting is depicted in Figure 1 where two



(a) Single-label learning trains a model to classify one label per instance, usually using labeled training data.

(b) Multi-label learning trains a model to assign multiple labels per instance, using their dependencies (Read et al. 2011).

(c) Single-label KA trains a student by learning from pre-trained teacher models to assign one label per instance (Luo et al. 2019; Ye et al. 2019; Thadajarassiri et al. 2021).

(d) Ours: Multi-label KA trains a student by learning from pre-trained multi-label teacher models to capture label dependencies used for assigning multiple labels per instance.

Figure 2: Comparison of related problems.

teachers are pre-trained to capture the dependencies among labels in their different specialized sets—{tree, garage, car} by Teacher 1 and {car, street, building} by Teacher 2. The student aims to handle all labels {tree, garage, car, street, building} by combining knowledge from both teachers.

Unfortunately, to-date no such multi-label KA methods exist. As shown in Figure 2c, existing KA works (Ye et al. 2019; Thadajarassiri et al. 2021) focus only on the simpler *single-label* classification, which disregards the crucial label dependency knowledge for the *multi-label* classification.

Problem Definition. We are the first to study the problem of *knowledge amalgamation for multi-label classification (KA-MLC).* As shown in Figure 1, our goal is to train a *student* model given only unlabeled data and a set of pre-trained multi-label models (*teachers*). These teachers may have different classifier chains-based architectures, the predominant methods to learn dependencies between labels (Chen et al. 2019). The student's aim is to accurately classify the labels in the union of all teachers' label sets by unifying the teachers' respective knowledge of label dependencies.

Challenges. Three main challenges arise for KA-MLC:

• *No labeled data*. Traditional methods for MLC require access to a huge amount of training data with ground truth labels. With only unlabeled data, conventional supervised methods are not applicable, thus necessitating the development of a novel solution not relying on human annotations.

• *Teacher disagreement*. Individual teachers may learn different knowledge as they are trained on their own private data. In some cases, teachers may disagree about a label, *e.g.*, it may be unclear whether there is a *car* in Figure 1. So Teacher 1 may predict positive while Teacher 2 predicts negative. A good solution must determine how to combine such contradictory predictions into one prediction.

• *Partially overlapping label sets.* Each teacher is pre-trained on a different subset of the labels to be learned by the student. This leads to incomplete dependency knowledge for some labels with respect to the student's task, *e.g.*, *garage* and *street* in Figure 1. Each is observed by only one teacher and their dependency knowledge has never been learned. However, an ideal student should still learn effectively the dependency between these *disjoint* labels across teachers.

Proposed Method. We propose **A**daptive KN**o**wledge **T**ransfer (ANT), the first method to solve the challenging KA-MLC problem. Our key idea is to train a student model from teachers that exchange their label dependency knowledge adaptively in order to model the global dependencies between all labels in the union of their label sets.

ANT unifies knowledge from teachers adaptively to each instance, overcoming the contradictory predictions between the teachers that may apply to some instances with unclear signals, *i.e.*, labels predicted positive by one teacher and negative by another. Since multi-label models are known to be used for *rejecting* instances with ambiguous signals (Hendrickx et al. 2021), the teachers commonly provide low probability to these labels. Regarding this principle, ANT is trained to trust the teacher that predicts positive as this teacher shows strong competence in utilizing its other labels to infer this positive prediction. Moreover, ANT facilitates the competent teacher (*Transferor*) to transfer its prediction to the other teachers (*Transferees*). These *transferees* are encouraged to revise their predictions by conditioning on the prediction of the more knowledgeable teacher. This way, ANT succeeds in utilizing label dependencies jointly among teachers to learn the global dependency knowledge for labels existing across teachers.

Contributions. Our contributions include the following:

• We define the open problem of knowledge amalgamation for multi-label classification (KA-MLC). The aim is to train a student that can handle all labels specialized across multiple multi-label teachers, using only unlabeled data.

• We propose Adaptive KNowledge Transfer (ANT), the first solution to this open problem. ANT trains a student by unifying predictions extracted from teachers, in part by facilitating the latter to exchange their knowledge.

• We demonstrate that ANT outperforms five state-of-the-art KA methods on eight datasets by achieving on average the best performance on four standard multi-label metrics.

Related Work

Knowledge Amalgamation (KA). KA (Shen et al. 2019a) builds on the *student-teacher* learning paradigm of Knowledge Distillation (Hinton, Vinyals, and Dean 2015), where a student model is trained to mimic one teacher model's predictions. KA advances beyond distillation by combining multiple teachers, each with different specialized tasks, into a student that is an expert on *all* the teachers' tasks.

As shown in Figure 2, existing KA works (Shen et al. 2019a; Luo et al. 2019; Ye et al. 2019; Shen et al. 2019b; Vongkulbhisal, Vinayavekhin, and Visentini-Scarzanella 2019; Thadajarassiri et al. 2021) study KA for single-label classification. Most approaches (Shen et al. 2019a; Luo et al. 2019; Vongkulbhisal, Vinayavekhin, and Visentini-Scarzanella 2019; Thadajarassiri et al. 2021) train a student to predict one class per instance from the union of all the teachers' classes. They do not consider informative dependencies between *labels*, which is essential in MLC. More importantly, these works cannot support the problem of KA-MLC that needs to identify multiple labels simultaneously.

Some works (Ye et al. 2019; Shen et al. 2019b) study multi-task classification that allows multiple labels to be assigned for each instance. However, these works treat each label as an independent single-label classification task. They overlook the label dependencies needed to solve the MLC, which is the key challenge of KA-MLC in integrating label dependency knowledge captured differently across teachers.

Moreover, their methods require the student and all teachers to have an identical number of layers in their architectures, which often does not hold in practice. Ye et al. (2019) trains a student by replacing each layer of each teacher with the student's corresponding layer and ensuring the teacher still makes the same prediction, even with the new layer. Similarly, Shen et al. (2019b) aligns the corresponding layers of the student and the teachers into a transfer bridge and maximizes their similarity.

Multi-Label Classification (MLC). MLC is the classification setting where multiple labels can correspond to the same instance simultaneously. Traditional approaches transform this problem into multiple binary classification tasks, one for each label. These methods (Tsoumakas and Katakis 2007; Godbole and Sarawagi 2004) fail to model dependencies between labels, a key requirement for successful MLC.

The best-known method for capturing label dependencies, *Classifier Chain* (CC) (Dembszynski et al. 2010; Dembczynski, Cheng, and Hüllermeier 2010; Read et al. 2011), has a long track record of successful use for challenging MLC tasks. CCs predict labels sequentially, conditioning each label prediction on previously predicted labels. Classic CCs require a predefined order of labels for their training, which is rarely available in practice.

Several recent works thus propose CCs that can be trained without a predefined label order, making them *order free* (Nam et al. 2017; Chen et al. 2018; Nam et al. 2019; Tsai and Lee 2020). These methods typically use Recurrent Neural Networks (RNNs) to predict labels one by one while modeling the transition between the predicted labels. This is achieved by feeding predicted labels back into the network at each step. Order-free CCs are currently the state-of-the-art solution to MLC, achieving strong performance in many impactful applications (Chen et al. 2019; You et al. 2020).

Many other works study MLC in various aspects such as handling new labels (Wang, Liu, and Tao 2020), detecting out-of-distribution instances (Wang et al. 2021), or learning from partially-labeled instances (Xie and Huang 2018). However, these works require clean labeled data while the KA-MLC assumes no labeled data are available.

Problem Formulation

In this paper, we study the new problem of knowledge amalgamation for multi-label classification (KA-MLC). In this setting, we are given unlabeled data, denoted as $\mathcal{X} = \{\mathbf{x}^i\}_{i=1}^n$ where $\mathbf{x}^i \in \mathbb{R}^d$ represents an instance with d features, and a set of m powerful pre-trained classifier chain based models (*teachers*), $\mathcal{T} = \{\mathbf{T}^t\}_{t=1}^m$. Each teacher specializes in solving a particular multi-label task for a set of ℓ^t distinct labels, denoted by the label set $\mathcal{Y}^t$. Thus, the predicted outputs for each instance $\mathbf{x}^i$ from each teacher $\mathbf{T}^t$ are $\hat{\mathcal{Y}}^{t,i} = \{\hat{y}_j^{t,i}\}_{y_j \in \mathcal{Y}^t}$ where $\hat{y}_j^{t,i} = 1$ (positive) if $\mathbf{T}^t$ predicts that the label y_j associates with instance $\mathbf{x}^i$ or 0 (negative) otherwise.

Our goal is to train a *student* model that accurately classifies $\mathbf{x}^i$ to its associated labels in the union of specialized labels of all teachers, $\mathcal{Y} = \{y_j\}_{j=1}^{\ell}$ where ℓ is the number of distinct labels. We note that $\mathcal{Y} = \bigcup_{t=1}^m \mathcal{Y}^t$. The student's outputs for the given $\mathbf{x}^i$ are thus $\hat{\mathcal{Y}}^i = \{\hat{y}_j^i\}_{j=1}^{\ell}$ where $\hat{y}_j^i \in \{0, 1\}$. We describe the rest of the paper in terms of one instance x^i and drop the superscript i hereafter.

State-of-the-art for MLC tend to model the joint dependencies between labels (Nam et al. 2019) by iteratively predicting each label using information from the previously predicted label, referred to as *Order-Free Classifier Chains (OFCC)* (Chen et al. 2018). Thus, we describe our approach in terms of OFCC-based teachers; however, with only slight modifications our method could be applied to other multi-label approaches that likewise model label dependencies.

Proposed Method: ANT

We propose **A**daptive K**N**owledge **T**ransfer (ANT) to solve the KA-MLC. ANT consists of three major components: (1) When teachers disagree on a label, the *Transfer Indicator (TI)* decides which teacher should transfer its prediction to which teacher; (2) the *Knowledge Transfer Module* revises

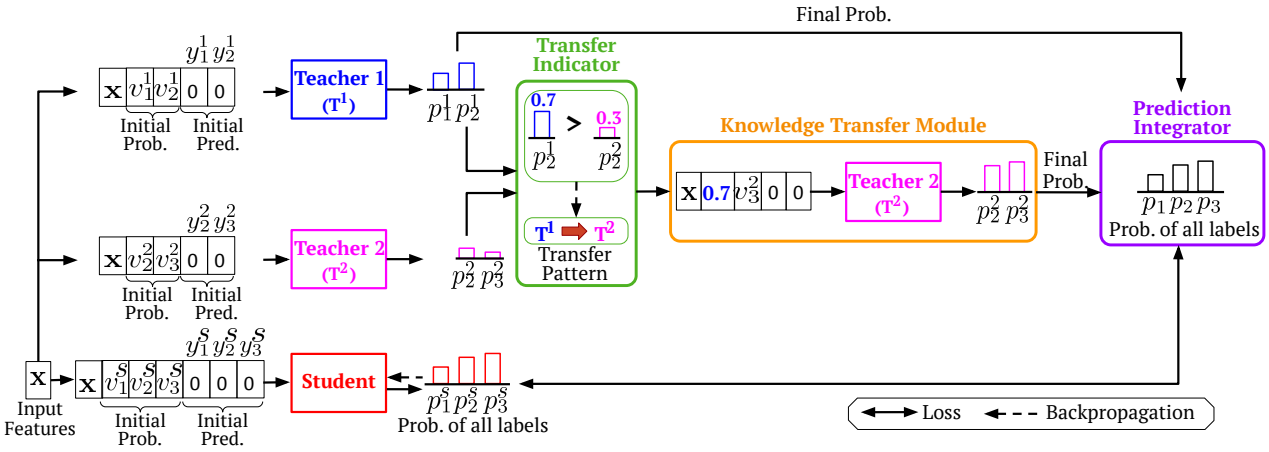


Figure 3: Overview of our proposed method, ANT. In this figure, one student is shown being trained to amalgamate the knowledge of two pre-trained teacher models. Input features are passed to each model during training.

a teacher's prediction, conditioned on the prediction of the more-competent teacher indicated by TI; (3) the *Prediction Integrator* combines all teachers' final predictions into one integrated prediction that is used to train the student.

Transfer Indicator (TI). TI first extracts the set of teacher relations $\mathcal{R}$, containing teacher-pairs and the shared label: $\mathcal{R} = \{\mathbf{r}_o\}_{o=1}^{r}$. Each $\mathbf{r}_o$ consists of two teachers and the label y_c that they specialize in common without regard for the teachers' order. For example, as shown in Figure 3, $\mathcal{R} = \{(\mathbf{T}^1, \mathbf{T}^2, y_2)\}$ since both $\mathbf{T}^1$ and $\mathbf{T}^2$ specialize on y_2.

Considering each instance $\mathbf{x}$ and $\mathbf{r}_o = (\mathbf{T}^m, \mathbf{T}^n, y_c)$, each teacher outputs the soft predictions or logits ($\mathcal{L}$) for its specialized labels that are passed through the sigmoid function (σ) to acquire their predicted probabilities ($\mathcal{P}$) as follows:

$$\mathcal{L}^m = \mathbf{T}^m(\mathbf{x}) \text{ and } \mathcal{P}^m = \sigma(\mathcal{L}^m) \quad (1)$$

$$\mathcal{L}^n = \mathbf{T}^n(\mathbf{x}) \text{ and } \mathcal{P}^n = \sigma(\mathcal{L}^n) \quad (2)$$

where $\mathcal{P}^t = \{p_j^t\}_{y_j \in \mathcal{Y}^t}$ and $p_j^t = P(y_j|\mathbf{x})$ which is the predicted probability of y_j provided by $\mathbf{T}^t$. The hard prediction is obtained by binarizing p_j^t with a threshold of 0.5.

In some cases, the input features $\mathbf{x}$ alone do not contain enough information to yield an agreed prediction between the teachers on the common label y_c, *i.e.*, one predicts positive while the other predicts negative. Fundamentally, the multi-label teachers commonly predict low probabilities for this label since multi-label models are known as a standard method for *rejecting* instances when observing a *novel* class (Hendrickx et al. 2021). Regarding this principle, the teacher that provides positive prediction evidently shows stronger knowledge achieved by utilizing the dependencies between y_c and the other labels it specializes on. Thus, to integrate the knowledge between teachers adaptively for any instance, TI indicates the more-competent teacher (*transferor*) to transfer its positive prediction of y_c toward the other teacher (*transferee*). For example, as depicted in Figure 3, p_2^1 yields positive prediction while p_2^2 yields negative prediction. Thus, TI indicates the transfer pattern to be $\mathbf{T}^1 \rightarrow \mathbf{T}^2$. This information is passed to the Knowledge Transfer Module to encourage the transferee teacher to revise its predictions.

Knowledge Transfer Module. For a given $\mathbf{r}_o = (\mathbf{T}^m, \mathbf{T}^n, y_c)$, assume TI indicates that $\mathbf{T}^m$ is the *transferor* and $\mathbf{T}^n$ is the *transferee*. The transferee $\mathbf{T}^n$ revises its predictions by conditioning on the prediction of the shared label y_c informed by the transferor $\mathbf{T}^m$. We show below that the prediction of the other labels specialized by $\mathbf{T}^n$ can benefit additional information provided by $\mathbf{T}^m$.

Analysis of Information Gain: Let $\mathcal{Y}^m$ and $\mathcal{Y}^n$ be the specialized label sets of the transferor $\mathbf{T}^m$ and the transferee $\mathbf{T}^n$, respectively. Assume that y_c is their shared label and y_c^* is its ground truth while $\hat{y}_c^m$ and $\hat{y}_c^n$ denote its predictions given by $\mathbf{T}^m$ and $\mathbf{T}^n$, respectively. $I(X;Y)$ represents the mutual information between any random variables X and Y.

As we assume that transferor has made a more accurate prediction for y_c, the mutual information shared between $\hat{y}_c^m$ and y_c^* is higher than the mutual information shared between $\hat{y}_c^n$ and y_c^*, which is formalized in Assumption 1 as follows:

Assumption 1 (A1): $I(y_c^*; \hat{y}_c^m) > I(y_c^*; \hat{y}_c^n)$

$$\textit{i.e.}, \exists \lambda \in (0,1),\ \lambda I(y_c^*; \hat{y}_c^m) = I(y_c^*; \hat{y}_c^n)$$

Additionally, we assume that $\hat{y}_c^m$ and $\hat{y}_c^n$ are not biased with respect to y_j^*, the ground truth for the label y_j that is another label that $\mathbf{T}^n$ specializes in. Thus, if the transferee contains λ (less) of the information between itself and y_c than the transferor does, then it likewise contains λ (less) information between itself and the information content of y_j that is independent from y_c, stated formally as follows:

Assumption 2 (A2): Let y_j^* be the ground truth for the label y_j that is specialized particularly by $\mathbf{T}^n$.

$$\lambda I(y_c^*; \hat{y}_c^m) = I(y_c^*; \hat{y}_c^n) \Rightarrow \lambda I(y_c^*; \hat{y}_c^m | y_j^*) = I(y_c^*; \hat{y}_c^n | y_j^*)$$

Theorem 1: Let A1, A2 hold. Then, $I(\hat{y}_c^m; y_j^*) > I(\hat{y}_c^n; y_j^*)$.

Proof. By applying the chain rule of mutual information:

$$I(y_c^*; \hat{y}_c^m) = I(y_c^*; \hat{y}_c^m | y_j^*) + I(y_j^*; \hat{y}_c^m)$$

$$I(y_c^*; \hat{y}_c^n) = I(y_c^*; \hat{y}_c^n | y_j^*) + I(y_j^*; \hat{y}_c^n)$$
$$I(y_c^*; \hat{y}_c^n) = I(y_c^*; \hat{y}_c^n | y_j^*) + I(y_j^*; \hat{y}_c^n)$$

Applying A1 and A2, we have

$$\lambda I(y_c^*; \hat{y}_c^m | y_j^*) + \lambda I(y_j^*; \hat{y}_c^m) = \lambda I(y_c^*; \hat{y}_c^m | y_j^*) + I(y_j^*; \hat{y}_c^n)$$
$$\lambda I(y_j^*; \hat{y}_c^m) = I(y_j^*; \hat{y}_c^n)$$

Since $\lambda \in (0, 1)$, $I(\hat{y}_c^m; y_j^*) > I(\hat{y}_c^n; y_j^*)$.

Theorem 1 states that the information between the ground truth for y_j, which is a label that the transferee $\mathbf{T}^n$ particularly specializes in, and the prediction of the shared label from the transferor ($\hat{y}_c^m$) is greater than it is between the ground truth for y_j and the transferee's prediction for the shared label ($\hat{y}_c^n$). Thus, we should use $\hat{y}_c^m$ to infer $\hat{y}_j^n$ and set the initial predicted probability for y_c for $\mathbf{T}^n$ by using its predicted probability from $\mathbf{T}^m$, *i.e.*, Equation 2 is revised to:

$$\mathcal{L}^{n'} = \mathbf{T}^n(\mathbf{x}) \text{ and } \mathcal{P}^{n'} = \sigma(\mathcal{L}^{n'}) \tag{3}$$

Prediction Integrator. We finally combine the probabilities from all teachers—some of which may have been revised according to Equations 1 and 3—to compute predictions for all labels in $\mathcal{Y}$. To obtain this final probability for each label y_j, ANT acquires the most confident prediction from all teachers that specialize on y_j. Let $\mathcal{B}_j$ denote a set of teachers that specialize on y_j and $\mathcal{C}_j$ denote a set of candidate probabilities of y_j provided by these teachers:

$$\mathcal{C}_j = \{p_j^t\}_{\mathbf{T}^t \in \mathcal{B}_j} \tag{4}$$

The final integrated probability for each label is obtained as:

$$P(y_j|\mathbf{x}) = \begin{cases} min(\mathcal{C}_j) & \text{if } \forall p_j^t \in \mathcal{C}_j, p_j^t <= 0.5 \\ max(\mathcal{C}_j) & \text{otherwise.} \end{cases} \tag{5}$$

Training a Student Model. We use an RNN with LSTMs to feed back the previously predicted labels into the model, allowing to learn each label conditioned on the other labels. The model is fed three input components at each time step including the input features ($\mathbf{x}$), and both the soft predictions or logits ($\mathcal{L}$) and the hard predictions ($\hat{\mathcal{Y}}$) from the previous time step. At the first time step, the initial hard predictions $\hat{\mathcal{Y}}_0$ are all set as negative while the initial soft predictions ($\mathcal{L}_0$) are set by passing the features $\mathbf{x}$ through a linear layer. Therefore, the initial input vector ($\mathbf{x}_0$) is:

$$\mathcal{L}_0 = W \cdot \mathbf{x} + b \text{ and } \hat{\mathcal{Y}}_0 = [0]^\ell \tag{6}$$
$$\mathbf{x}_0 = [\mathbf{x}, \mathcal{L}_0, \hat{\mathcal{Y}}_0] \tag{7}$$

where W and b are learnable parameters.

For the time step k, $\mathcal{L}_k$ is updated using the three input components together with the previous hidden state as:

$$\mathbf{x}_k = [\mathbf{x}, \mathcal{L}_{k-1}, \hat{\mathcal{Y}}_{k-1}] \quad \text{and} \quad \mathcal{L}_k = LSTM_\theta(\mathbf{x}_k, h_{k-1}) \tag{8}$$

where $LSTM$ denotes the entire process of an LSTM model, θ denotes all parameters for such the LSTM, and h_{k-1} denotes its previous hidden state.

Dataset	Domain	# Instances	Avg. # Labels	# Unique Labels	# Unique Label Sets
EMOTIONS	Media	593	1.87	6	27
SCENE	Media	2,407	1.07	6	15
YELP	Text	10,806	1.64	8	118
YEAST	Biology	2,417	4.24	14	198
BIRD	Media	645	1.01	19	133
TMC	Text	28,596	2.16	22	1,341
GENBASE	Biology	662	1.25	27	32
MEDICAL	Text	978	1.25	45	94

Table 1: Details of 8 benchmark datasets used for evaluation

To obtain $\hat{\mathcal{Y}}_k$, all positive labels in $\hat{\mathcal{Y}}_{k-1}$ are removed from the candidate labels for prediction at the current step. Then the probabilities for all candidate labels are computed by passing the logits in Equation 8 through the sigmoid function. The label with the highest probability is predicted to be positive at this time step. Let z_k denote such the label that gets positive prediction at time step k. Thus, the joint probability of all labels at the last time step is:

$$Q(\mathcal{Y}|\mathbf{x}) = P(z_1|\mathbf{x}) \cdot \prod_{j=2}^{\ell} P(z_j|\mathbf{x}, z_1, \ldots, z_{j-1}) \tag{9}$$

The final logit for each label is obtained by carrying its logit from the time step that its hard prediction is selected to be positive. We denote the final logits for all labels in $\mathcal{Y}$ as $\mathcal{L}^s = \{v_1, \ldots, v_\ell\}$, passed through the sigmoid function to obtain the predicted probabilities for each label as:

$$\mathcal{P}^s = \sigma(\mathcal{L}^s) \text{ where } \mathcal{P}^s = \{p_j^s\}_{j=1}^{\ell} \text{ and } p_j^s = Q(y_j|\mathbf{x}) \tag{10}$$

We note that $Q(y_j|\mathbf{x})$ denotes the predicted probability of the label y_j learned by the student.

Finally, the student is trained to update θ iteratively by minimizing binary cross entropy between the predicted probability $Q(y_j|\mathbf{x})$ in Equation 10 and the integrated probability $P(y_j|\mathbf{x})$ in Equation 5 as follows:

$$J(\theta) = -\sum_{j=1}^{\ell} \Big(P(y_j|\mathbf{x}) \log(Q(y_j|\mathbf{x})) + (1 - P(y_j|\mathbf{x})) \log(1 - Q(y_j|\mathbf{x})) \Big). \tag{11}$$

Experiments

Datasets. We conduct experiments on eight established benchmark datasets for evaluating multi-label classifiers including *EMOTIONS* (Trohidis et al. 2008), *SCENE* (Boutell et al. 2004), *YELP* (Sajnani et al. 2012), *YEAST* (Elisseeff and Weston 2001), *BIRD* (Briggs et al. 2013), *TMC* (Srivastava and Zane-Ulman 2005), *GENBASE* (Diplaris et al. 2005), and *MEDICAL* (Pestian et al. 2007). The number of instances, average labels per instance, unique labels and label sets per dataset are shown in Table 1.

Compared Methods. We compare ANT to five state-of-the-art methods: (1) ***Baseline (BL)*** combines hard predictions

Methods	S-Loss↓	H-Loss↓	F1 Micro↑	F1 Macro↑	Rank↓
EMO Dataset					
BL	.9500±.0273	.3726±.0359	.4351±.0621	.4539±.0492	3.8
AKA	.9786±.0273	.4346±.0611	.3184±.0848	.2069±.1308	5.8
FKA	.9714±.0404	.4417±.0840	.3812±.0757	.2662±.0951	5.3
CFL	.9357±.0273	.3464±.0416	.4372±.0815	.4193±.1194	3.0
TC	**.9143**±.0234	.3452±.0228	.4366±.0542	.4548±.0354	2.0
ANT (Ours)	.9286±.0286	**.3345**±.0211	**.4545**±.0398	**.4559**±.0177	**1.3**
SCENE Dataset					
BL	.6615±.0273	.1953±.0359	.4592±.0621	.4490±.0492	3.0
AKA	.8490±.0273	.2416±.0611	.4038±.0848	.4092±.1308	5.0
FKA	1.000±.0404	.5680±.0840	.2784±.0757	.2339±.0951	6.0
CFL	.6684±.0273	.1933±.0416	.4552±.0815	.4450±.1194	3.8
TC	.6424±.0234	.1869±.0228	.4645±.0542	.4459±.0354	2.3
ANT (Ours)	**.6077**±.0286	**.1765**±.0211	**.4840**±.0398	**.4675**±.0177	**1.0**
YELP Dataset					
BL	**.9723**±.0070	.3297±.0087	.4151±.0121	.3945±.0080	2.3
AKA	.9796±.0064	.3698±.0198	.3925±.0186	.3725±.0165	4.3
FKA	1.000±.0000	.5807±.0722	.3540±.0430	.3031±.0793	6.0
CFL	.9769±.0133	.3290±.0119	.3778±.0175	.3635±.0143	4.3
TC	.9734±.0039	.3228±.0050	.3921±.0168	.3779±.0109	3.0
ANT (Ours)	.9730±.0030	**.3177**±.0063	**.4217**±.0113	**.4044**±.0060	**1.3**
YEAST Dataset					
BL	.9983±.0035	.4135±.1233	.4733±.0614	.3345±.0245	4.5
AKA	.9966±.0040	**.3409**±.0347	.4749±.0167	**.3630**±.0106	2.3
FKA	1.000±.0000	.5390±.0640	.3884±.0361	.3047±.0429	5.8
CFL	**.9948**±.0066	.3514±.0119	.4792±.0081	.3200±.0107	3.0
TC	1.000±.0000	.3487±.0083	.4917±.0050	.3383±.0146	3.0
ANT (Ours)	**.9948**±.0066	.3553±.0112	**.5073**±.0061	.3535±.0091	**2.0**
BIRD Dataset					
BL	.5855±.0543	.0630±.0129	.0000±.0000	.0000±.0000	4.5
AKA	1.000±.0000	.3850±.0621	.0928±.0576	.0682±.0276	3.5
FKA	1.000±.0000	.5613±.0339	**.0937**±.0295	**.0779**±.0159	3.3
CFL	**.5592**±.0584	.0575±.0110	.0000±.0000	.0000±.0000	3.3
TC	**.5592**±.0584	.0599±.0093	.0114±.0228	.0038±.0075	3.0
ANT (Ours)	**.5592**±.0584	**.0561**±.0100	.0132±.0263	.0053±.0106	**2.0**
TMC Dataset					
BL	.9997±.0003	.1877±.0097	.0910±.0225	.0733±.0055	3.3
AKA	1.000±.0000	.4075±.0166	.1967±.0023	**.1874**±.0069	3.3
FKA	1.000±.0000	.5477±.1077	**.2045**±.0068	.1459±.0077	3.5
CFL	.9997±.0006	.1810±.0015	.0761±.0025	.0713±.0053	4.0
TC	.9997±.0003	.1820±.0036	.0793±.0048	.0725±.0056	3.8
ANT (Ours)	**.9994**±.0005	**.1784**±.0015	.0825±.0033	.0746±.0013	**2.3**
GENBASE Dataset					
BL	**.9295**±.0385	.0833±.0077	.0734±.0261	.0370±.0000	2.5
AKA	1.000±.0000	.3991±.0642	**.1748**±.0299	**.1853**±.0307	3.0
FKA	1.000±.0000	.6769±.1180	.0808±.0136	.0600±.0047	3.8
CFL	**.9295**±.0385	.0845±.0097	.0728±.0268	.0370±.0000	3.5
TC	**.9295**±.0385	.0836±.0091	.0734±.0265	.0370±.0000	3.0
ANT (Ours)	**.9295**±.0385	**.0829**±.0083	.0740±.0266	.0370±.0000	**2.0**
MEDICAL Dataset					
BL	**.9957**±.0086	.0519±.0029	.0067±.0134	.0078±.0156	3.5
AKA	1.000±.0000	.4013±.0300	**.0655**±.0037	**.0832**±.0098	2.5
FKA	1.000±.0000	.5403±.0166	.0574±.0046	.0373±.0063	3.3
CFL	1.000±.0000	.0467±.0053	.0114±.0146	.0041±.0049	3.8
TC	1.000±.0000	**.0432**±.0037	.0046±.0092	.0005±.0011	4.0
ANT (Ours)	**.9957**±.0086	.0458±.0018	.0119±.0151	.0089±.0117	**2.3**

Table 2: Compared performance (mean±std) on eight benchmark datasets. Subset Loss and Hamming Loss are abbreviated as S-Loss and H-Loss, respectively. Rank shows overall performance across all metrics. ↑/↓ indicates the larger/smaller the better.

from all teachers using majority voting, assuming positive for even votes. (2) ***AKA*** (Shen et al. 2019b) trains an individual student for each label, combined later into one final student. Corresponding layers of the teachers and student are trained to be similar. (3) ***FKA*** (Ye et al. 2019) replaces the teacher's layer with the corresponding layer of the student and then minimizes the difference between the output of the modified teacher and its original output. (4) ***CFL*** (Luo et al. 2019) imitates the teachers' logits and their final layers mapped into a common space in which the student's final layer is trained to be similar. (5) ***TC*** (Thadajarassiri et al. 2021) imitates the weighted sum of teachers' logits. We use equal weights since our data are unlabeled.

Implementation Details. For each dataset, we randomly select 70% of the instances for the teachers and use the remaining 30% for experiments with students. In main experiments, we set the number of labels that overlap between teachers such that teachers hold roughly an equal number of their own specialized labels and roughly the other half of their labels overlap with other teachers. The choice of labels is random. We run four replications for each experiment with different random seeds, using 75% of the 30% left-out data to train the student models, and use the remaining 25% for testing. All code, datasets, and experimental details are made publicly available at https://github.com/jida-thada/ANT.

Experimental Results

We measure performance using four standard multi-label metrics: Subset Loss, Hamming Loss, F1 Micro, and F1 Macro. The averaged rank across all metrics is reported to compare their overall performance; ranked-1 indicates the best performance.

Effectiveness of ANT across Benchmark Datasets. First, we demonstrate that ANT outperforms the five alternative methods by successfully leveraging label dependency knowledge captured differently between teachers. For this experiment, we train a student from two teachers trained on different data—from each original dataset—that cover different subsets of labels, *i.e.*, each teacher is trained to capture the dependency knowledge differently regarding its particular label set. For each dataset, teachers learn from an equal number of labels with half of their labels overlapping.

Methods	S-Loss↓	H-Loss↓	F1 Micro↑	F1 Macro↑	Rank↓
50% of labels shared between teachers					
BL	.780±.021	.263±.010	.292±.012	.305±.021	4.3
AKA	.884±.062	.259±.055	**.375**±.042	**.382**±.059	2.8
FKA	1.000±.000	.544±.067	.246±.056	.191±.046	6.0
CFL	.785±.020	.251±.008	.295±.015	.303±.024	3.8
TC	.774±.023	**.242**±.009	.299±.010	.308±.014	2.3
ANT	**.766**±.017	.253±.005	.305±.009	.309±.021	**2.0**
25% of labels shared between teachers					
BL	.661±.027	.195±.036	.459±.062	.449±.049	3.0
AKA	.849±.027	.242±.061	.404±.085	.409±.131	5.0
FKA	1.000±.040	.568±.084	.278±.076	.234±.095	6.0
CFL	.668±.027	.193±.042	.455±.082	.445±.119	3.8
TC	.642±.023	.187±.023	.465±.054	.446±.035	2.3
ANT	**.608**±.029	**.177**±.021	**.484**±.040	**.468**±.018	**1.0**
0% of labels shared between teachers					
BL	.498±.026	.122±.012	.654±.041	.660±.046	3.0
AKA	.891±.055	.274±.024	.385±.038	.398±.057	5.0
FKA	1.000±.000	.532±.073	.301±.025	.298±.038	6.0
CFL	.517±.037	.120±.010	.658±.028	.662±.032	2.8
TC	.505±.041	.123±.013	.651±.033	.662±.042	3.3
ANT	**.439**±.031	**.106**±.011	**.685**±.032	**.690**±.037	**1.0**

Table 3: Results from varying the number of labels shared between teachers observed on the SCENE dataset.

Methods	S-Loss↓	H-Loss↓	F1 Micro↑	F1 Macro↑	Rank↓
3 Teachers					
BL	.930±.013	.299±.013	.479±.011	.375±.011	3.5
AKA	.980±.006	.370±.020	.395±.015	.382±.006	4.3
FKA	.998±.003	.495±.062	.349±.014	.308±.011	6.0
CFL	.921±.011	.283±.003	.473±.007	.365±.010	3.5
TC	.920±.006	**.280**±.003	.481±.004	.372±.008	2.3
ANT	**.918**±.005	.286±.004	**.488**±.010	**.382**±.010	**1.5**
4 Teachers					
BL	.971±.007	.364±.005	.373±.007	.279±.011	3.8
AKA	.980±.006	.379±.016	**.385**±.011	**.371**±.013	3.0
FKA	.998±.003	.506±.039	.341±.041	.306±.043	5.0
CFL	.968±.009	.349±.007	.358±.008	.265±.004	3.8
TC	**.962**±.006	**.341**±.003	.361±.005	.264±.004	3.0
ANT	.967±.005	.355±.005	.373±.003	.281±.004	**2.5**
5 Teachers					
BL	.977±.005	.374±.003	.373±.009	.281±.010	3.8
AKA	.990±.004	.390±.010	**.397**±.011	**.384**±.008	3.0
FKA	.997±.003	.462±.028	.340±.035	.284±.014	5.3
CFL	.964±.010	**.338**±.014	.349±.008	.255±.015	3.5
TC	**.956**±.011	.342±.006	.358±.004	.267±.009	3.0
ANT	.970±.005	.365±.005	.382±.002	.289±.008	**2.5**

Table 4: Results of amalgamating knowledge across many teachers observed on the YELP dataset.

The results on eight datasets are shown in Table 2. ANT consistently achieves the highest rank averaged across all metrics across the board while the second-best methods alternate between BL, AKA, and TC depending on different datasets. We notice that BL performs quite good comparing to the other methods. This is because we apply the same core rationale in the proposed ANT to trust more on the teacher that predicts positive when there are contradictory predictions between the teachers. As expected, ANT is the strongest method particularly for the Hamming Loss and the Subset Loss, which are core metrics for multi-label learning. It achieves the best performance for six out of eight datasets for both metrics. This demonstrates ANT's accuracy for both the individual labels and the entire label sets.

ANT Resolves Teacher Disagreement. To investigate the case where teachers learn vastly different knowledge from each other, we follow other recent work (Thadajarassiri et al. 2021) by varying the number of shared labels between teachers. Using the SCENE dataset, where most methods perform their best across all metrics, we vary the proportion of labels shared between two teachers from 50% (half) to 0% (none).

The result in Table 3 show that as teachers share fewer labels, the resulting students become more effective. With fewer shared labels, the teachers have less opportunity to provide contradictory feedback to the student. All in all, ANT achieves the top average rank across the board on average. This shows that ANT extracts predictions from heterogeneous sources more reliably than the state-of-the-art. ANT shows especially impressive on the Subset Loss which is the only metric that measures label-set performance.

Bridging Label Dependencies across Multiple Teachers. Finally, we explore the more challenging case of amalgamating many teachers which naturally creates several potential unobserved dependencies between labels that each label exists across teachers. We conduct experiments using 3, 4, and 5 teachers on the YELP dataset that contains many instances allowing us to train more independent teachers.

As shown in Table 4, once again, ANT achieves the highest rank for all settings. This indicates that ANT not only learns the dependency knowledge from each particular teacher but also effectively infers unobserved dependencies between labels that exist across teachers. This is achieved by allowing the transferee teacher to revise its predictions based on the predictions of more-competent teachers.

Conclusion

We propose **A**daptive K**N**owledge **T**ransfer (ANT), the first solution for knowledge amalgamation for multi-label classification (KA-MLC). The goal is to train a student model that can capture the dependencies among all labels across all teachers. To achieve this, ANT encourages each teacher to revise their prediction based on the knowledge from the more competent teachers. This way, ANT succeeds in utilizing label dependencies jointly across teachers so as to infer the overall global dependencies between labels in the union of the teachers' label sets. Our comprehensive experimental study on eight real-world datasets demonstrates that ANT significantly outperforms state-of-the-art alternatives by achieving the best averaged rank across numerous standard multi-label metrics for all datasets.

Acknowledgements

This research was supported by NSF under IIS-1910880, CSSI-2103832, and NRT-HDR-1815866. We also thank all members of the DAISY research group at WPI.

References

Boutell, M. R.; Luo, J.; Shen, X.; and Brown, C. M. 2004. Learning multi-label scene classification. *Pattern recognition*, 37(9): 1757–1771.

Briggs, F.; Yonghong, H.; Raich, R.; et al. 2013. New methods for acoustic classification of multiple simultaneous bird species in a noisy environment. In *IEEE International Workshop on Machine Learning for Signal Processing*, 1–8.

Chen, S.-F.; Chen, Y.-C.; Yeh, C.-K.; and Wang, Y.-C. F. 2018. Order-free rnn with visual attention for multi-label classification. In *Proceedings of AAAI*, volume 32, 6714–6721.

Chen, Z.-M.; Wei, X.-S.; Wang, P.; and Guo, Y. 2019. Multi-label image recognition with graph convolutional networks. In *Proceedings of CVPR*, 5177–5186.

Dembczynski, K.; Cheng, W.; and Hüllermeier, E. 2010. Bayes optimal multilabel classification via probabilistic classifier chains. In *Proceedings of ICML*, 279–286.

Dembszynski, K.; Waegeman, W.; Cheng, W.; and Hüllermeier, E. 2010. On label dependence in multilabel classification. In *ICML Workshop on Learning from Multi-label data*, 5–12.

Diplaris, S.; Tsoumakas, G.; Mitkas, P. A.; and Vlahavas, I. 2005. Protein classification with multiple algorithms. In *Proceedings of Panhellenic Conference on Informatics*, 448–456. Springer.

Elisseeff, A.; and Weston, J. 2001. A kernel method for multi-labelled classification. In *Proceedings of NeurIPS*, volume 14, 681–687.

Godbole, S.; and Sarawagi, S. 2004. Discriminative methods for multi-labeled classification. In *Pacific-Asia conference on knowledge discovery and data mining*, 22–30. Springer.

Hendrickx, K.; Perini, L.; Van der Plas, D.; Meert, W.; and Davis, J. 2021. Machine Learning with a Reject Option: A survey. *arXiv preprint arXiv:2107.11277*.

Hinton, G.; Vinyals, O.; and Dean, J. 2015. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*.

Luo, S.; Wang, X.; Fang, G.; Hu, Y.; Tao, D.; and Song, M. 2019. Knowledge amalgamation from heterogeneous networks by common feature learning. In *Proceedings of IJCAI*, 3087–3093.

Nam, J.; Kim, Y.-B.; Mencia, E. L.; Park, S.; Sarikaya, R.; and Fürnkranz, J. 2019. Learning context-dependent label permutations for multi-label classification. In *Proceedings of ICML*, 4733–4742.

Nam, J.; Mencía, E. L.; Kim, H. J.; and Fürnkranz, J. 2017. Maximizing subset accuracy with recurrent neural networks in multi-label classification. In *Proceedings of NeurIPS*, 5413–5423.

Pestian, J.; Brew, C.; Matykiewicz, P.; Hovermale, D. J.; Johnson, N.; Cohen, K. B.; and Duch, W. 2007. A shared task involving multi-label classification of clinical free text. In *Proceedings of Biological, translational, and clinical language processing*, 97–104.

Read, J.; Pfahringer, B.; Holmes, G.; and Frank, E. 2011. Classifier chains for multi-label classification. *Machine learning*, 85(3): 333–359.

Sajnani, H.; Saini, V.; Kumar, K.; Gabrielova, E.; Choudary, P.; and Lopes, C. 2012. Classifying yelp reviews into relevant categories. *Mondego Group, Univ. California Press, Berkeley, CA USA, Tech. Rep*.

Shen, C.; Wang, X.; Song, J.; Sun, L.; and Song, M. 2019a. Amalgamating knowledge towards comprehensive classification. In *Proceedings of AAAI*, 3068–3075.

Shen, C.; Xue, M.; Wang, X.; Song, J.; Sun, L.; and Song, M. 2019b. Customizing student networks from heterogeneous teachers via adaptive knowledge amalgamation. In *Proceedings of ICCV*, 3504–3513.

Srivastava, A. N.; and Zane-Ulman, B. 2005. Discovering recurring anomalies in text reports regarding complex space systems. In *Proceedings of IEEE aerospace*, 3853–3862.

Thadajarassiri, J.; Hartvigsen, T.; Kong, X.; and Rundensteiner, E. 2021. Semi-Supervised Knowledge Amalgamation for Sequence Classification. In *Proceedings of AAAI*, volume 35, 9859–9867.

Trohidis, K.; Tsoumakas, G.; Kalliris, G.; Vlahavas, I. P.; et al. 2008. Multi-label classification of music into emotions. In *Proceedings of ISMIR*, volume 8, 325–330.

Tsai, C.-P.; and Lee, H.-Y. 2020. Order-free learning alleviating exposure bias in multi-label classification. In *Proceedings of AAAI*, volume 34, 6038–6045.

Tsoumakas, G.; and Katakis, I. 2007. Multi-label classification: An overview. *Data Warehousing and Mining (IJDWM)*, 3(3): 1–13.

Vens, C.; Struyf, J.; Schietgat, L.; Džeroski, S.; and Blockeel, H. 2008. Decision trees for hierarchical multi-label classification. *Machine learning*, 73(2): 185.

Vongkulbhisal, J.; Vinayavekhin, P.; and Visentini-Scarzanella, M. 2019. Unifying heterogeneous classifiers with distillation. In *Proceedings of CVPR*, 3175–3184.

Wang, H.; Liu, W.; Bocchieri, A.; and Li, Y. 2021. Can multi-label classification networks know what they don't know? In *Proceedings of NeurIPS*, volume 34, 29074–29087.

Wang, J.; Yang, Y.; Mao, J.; Huang, Z.; Huang, C.; and Xu, W. 2016. Cnn-rnn: A unified framework for multi-label image classification. In *Proceedings of CVPR*, 2285–2294.

Wang, Z.; Liu, L.; and Tao, D. 2020. Deep streaming label learning. In *International Conference on Machine Learning*, 9963–9972. PMLR.

Xie, M.-K.; and Huang, S.-J. 2018. Partial multi-label learning. In *Proceedings of AAAI*, volume 32.

Yang, P.; Sun, X.; Li, W.; Ma, S.; Wu, W.; and Wang, H. 2018. SGM: Sequence Generation Model for Multi-label

Classification. In *Proceedings of Computational Linguistics*, 3915–3926.

Ye, J.; Wang, X.; Ji, Y.; Ou, K.; and Song, M. 2019. Amalgamating filtered knowledge: learning task-customized student from multi-task teachers. In *Proceedings of IJCAI*, 4128–4134.

You, R.; Guo, Z.; Cui, L.; Long, X.; Bao, Y.; and Wen, S. 2020. Cross-modality attention with semantic graph embedding for multi-label classification. In *Proceedings of AAAI*, volume 34, 12709–12716.

Zhang, M.-L.; and Zhang, K. 2010. Multi-label learning by exploiting label dependency. In *Proceedings of ACM SIGKDD*, 999–1008.