

Joint Learning of Hyperbolic Label Embeddings for Hierarchical Multi-label Classification

Soumya Chatterjee^{1*} Ayush Maheshwari^{1*}
Ganesh Ramakrishnan¹ Saketha Nath Jagaralpudi²

¹ Indian Institute of Technology Bombay

² Indian Institute of Technology Hyderabad

{soumya, ayusham, ganesh}@cse.iitb.ac.in, saketha@iith.ac.in

Abstract

We consider the problem of multi-label classification, where the labels lie in a hierarchy. However, unlike most existing works in hierarchical multi-label classification, we do not assume that the label-hierarchy is known. Encouraged by the recent success of hyperbolic embeddings in capturing hierarchical relations, we propose to jointly learn the classifier parameters as well as the label embeddings. Such a joint learning is expected to provide a two-fold advantage: i) the classifier generalises better as it leverages the prior knowledge of existence of a hierarchy over the labels, and ii) in addition to the label co-occurrence information, the label-embedding may benefit from the manifold structure of the input datapoints, leading to embeddings that are more faithful to the label hierarchy. We propose a novel formulation for the joint learning and empirically evaluate its efficacy. The results show that the joint learning improves over the baseline that employs label co-occurrence based pre-trained hyperbolic embeddings. Moreover, the proposed classifiers achieve state-of-the-art generalization on standard benchmarks. We also present evaluation of the hyperbolic embeddings obtained by joint learning and show that they represent the hierarchy more accurately than the other alternatives. The source code of the paper is available [here](#).

1 Introduction

The problem of multi-label text classification is well known and extensively studied in literature (McCallum, 1999; Yang et al., 2009; Liu et al., 2017). The fundamental assumption is that a document is associated with multiple labels from a fixed vocabulary of labels. Often, these labels are organised in a hierarchical structure. For ex. consider a sample headline from the NYT (New York

Times) corpus “Voice Recognition Is Improving, but Don’t Stop the Elocution Lessons” for which labels are “Top/News/Technology”. Here, labels are arranged in a hierarchy, hereafter referred as label hierarchy. We undertake the task of labelling documents with classes that are hierarchically organised; this problem is popularly known as *hierarchical multi-label text classification* (HMC). HMC methods have found several applications in online advertising systems (Agrawal et al., 2013), bio-informatics (Peng et al., 2016; Triguero and Vens, 2016), text classification (Rousu et al., 2006; Mao et al., 2019).

The main challenge in HMC is in modelling classification of the document into a large, imbalanced and structured output space. In HMC, the label taxonomy is a partially ordered set (L, \prec) where L is a finite set of all class labels. Relation \prec refers to *is-a* relationship between labels, which is asymmetric, anti-reflexive and transitive (Silla and Freitas, 2011).

Hierarchical structures can provide important insights for learning and classification tasks. However, explicit knowledge of hierarchy is not available in several domains, for instance, extreme classification datasets (Bhatia et al., 2016). In this paper, we consider the problem of structured prediction from unstructured text, in which label hierarchy is not known *a priori*. We infer hierarchies from classification judgements on the outputs that are readily available. We focus on discovering relationships between the labels in a hyperbolic space, which has natural capacity to encode hierarchical structures.

In our approach, HIDDEN (Hyperbolic label embeddings for hierarchical multi-label classification), the labels are represented in a hyperbolic space to help respect their latent hierarchical organisation. We use this intuition to learn label embeddings for HMC without explicit supervision

*Equal contribution

on the label hierarchy.

Apart from employing hyperbolic embeddings, another key aspect of our methodology is that the parameters of the classifier as well as of the label embedding are learnt jointly. We next explain the advantage in doing so. In the absence of any partial information regarding the hierarchy, label embeddings are typically learnt using the weak supervision available in label co-occurrences (Nickel and Kiela, 2017, 2018). This weak form of supervision can be complemented if the label embedding learning is also aware of the manifold structure of the input (documents). For *e.g.*, similar documents may have similar labels, *etc.* Such a strengthening is possible only if learning happens in a joint fashion. Moreover, the generalization of the classifier also improves because of the improved embeddings (and vice-versa). Our contributions can be summarised as follows:

1. We present an approach HIDDEN, that models the implicit hierarchical organisation of labels for improved classification. It leverages properties of hyperbolic geometry to help learn embeddings for the hierarchically organised labels.
2. We present a novel formulation for jointly learning the parameters of the classifier as well as the label embedding, which can be trained solely using the supervision from the training data, and without using any explicit information regarding the label hierarchy.
3. We evaluate HIDDEN on real-world as well as synthetic datasets and show:
 - (a) significant improvement over classical multi-label classification methods as well as baselines that employ hyperbolic label embeddings learnt in isolation solely based on label co-occurrence information
 - (b) HIDDEN sometimes generalizes even better than state-of-the-art hierarchical multi-label classifiers that have complete access to the true label hierarchy
 - (c) label embeddings learnt using the joint optimisation approach correlate better with the ground truth than other alternatives.

2 Related Work

Several conventional classification methods are capable of handling classification in multi-label settings. However, relatively fewer of these are designed to incorporate the possibly hierarchical organisation of the class labels. These include both

traditional methods (Gopal and Yang, 2013; Lewis et al., 2004) as well as deep learning methods (Johnson and Zhang, 2015; Peng et al., 2018) across varied domains such as news articles, web content, *etc.* Some approaches (Bairi et al., 2015, 2016) have also attempted to identify a subset of class labels from the classification hierarchy that effectively *represents* most instances from the training dataset.

Traditional or flat classification approaches typically perform prediction assuming that all the classes are independent of each other, ignoring the class hierarchy. Whereas ‘local’ classification approaches (Koller and Sahami, 1997; Cesa-Bianchi et al., 2006) train a set of classifiers at each level of the hierarchy. However, it has also been argued (Cerri et al., 2011) that it is impractical to train separate classifiers at each level. On the other hand, ‘global’ approaches (Silla Jr and Freitas, 2009; Wang et al., 2001) train a single classifier that factors in the complete class hierarchy, while often also explicitly factoring in the label-label correlation (Kulkarni et al., 2018). Unlike the local approach, ‘global’ approaches do not suffer from the error propagation problem, although they are prone to under-fit by not considering local information in the hierarchy.

Some recent papers have proposed a mix of local and global approaches for HMC. Wehrmann et al. (2018) propose an objective that leverages both local and global information while introducing global hierarchical violation penalty. Mao et al. (2019) employ a reinforcement learning framework to learn a label assignment policy. They model HMC as a *markov decision process*, wherein, the agent takes an action of label assignment on the tree hierarchy and receives scalar rewards as feedback for the actions. Chen et al. (2019) embed both document and label hierarchy in the same hyperbolic space and use interactions between these embeddings for HMC. Our approach differs from these in two important ways: (i) we embed only labels into the hyperbolic space and (ii) label hierarchy is not known *a priori* - all we assume is that there is some hidden hierarchy.

Recently, the use of hyperbolic geometry has been found to be promising in machine learning and network sciences to model data with latent hierarchies. Krioukov et al. (2010) showed that properties of complex networks, namely *heterogeneous degree distribution* and *strong clustering*, naturally manifest in hyperbolic geometry. They

showed that if a network has some heterogeneous degree distribution and metric structure, the network can be mapped effectively to the hyperbolic space (since euclidean distance has limitations in approximating the distance between nodes in a tree). Gromov (1987) have shown that any finite tree structure can be embedded into a finite hyperbolic space while preserving the distance between nodes. Nickel and Kiela (2017) learnt hierarchical representations of symbolic data by embedding them into an n -dimensional Poincaré ball by leveraging the distance property of hyperbolic spaces. Instead of relying on the true hierarchy to learn embeddings, Nickel and Kiela (2018) inferred hierarchies from real-valued similarity scores using the Lorentz model of hyperbolic geometry. We used a similar formulation in our model HIDDEN to build our HMC model by leveraging the co-occurrence count of labels for each document, but additionally (and more importantly), in a joint manner, learn the parameters of the classifier.

3 Hyperbolic Geometry & the Poincaré Model

In this section, we give an overview of hyperbolic geometry and the Poincaré model for embedding in hyperbolic spaces (Nickel and Kiela, 2017). A hyperbolic space is a non-Euclidean Riemannian manifold of constant negative curvature. Though there are several fundamental differences between the Euclidean and the hyperbolic geometry, the most interesting characteristic of hyperbolic spaces is their ability to naturally represent hierarchical relations (Krioukov et al., 2010). In the Poincaré ball model, which is one of the standard models for hyperbolic geometry, the Euclidean distances between equidistant points, according to the inherent manifold metric d , fall exponentially as one moves from origin towards the surface of the ball. This interesting property is the key for enabling learning of continuous embeddings of hierarchies. For example, one can imagine root node of hierarchy at origin and leaf nodes near the ball’s surface. Then, this model can easily accommodate exponentially growing number of equidistant siblings at deeper levels of the hierarchy. Whereas, such an accommodation is not possible using Euclidean geometry. Below we provide some details of this model.

Let $\mathcal{B}^n = \{x \in \mathbb{R}^n \mid \|x\| < 1\}$ be the open n -dimensional unit ball, where $\|\cdot\|$ is the Euclidean 2 norm. The Poincaré ball model is

a Riemannian Manifold (\mathcal{B}^n, g_x) , the open unit ball equipped with the Riemannian metric tensor $g_x = \left(\frac{2}{1-\|x\|^2}\right)^2 g^E$, where $x \in \mathcal{B}^d$ and g^E is the Euclidean metric tensor. The geodesic distance between two points $u, v \in \mathcal{B}^d$ is given as

$$d(u, v) = \operatorname{arcosh} \left(1 + 2 \frac{\|u - v\|^2}{(1 - \|u\|^2)(1 - \|v\|^2)} \right) \quad (1)$$

Given any $x \in \mathbb{R}^n$, one can show that $\frac{x}{1 + \sqrt{1 + \|x\|_2^2}}$ always lies in the Poincaré ball (refer Appendix for detailed explanation).

4 Problem Formulation and Approach

In this section, we present present details of our model, training, as well as inference.

4.1 Problem Formulation

Here we consider an interesting special case of multi-label classification. The training data is of the form: $\mathcal{D} = \{(D_1, y_1), (D_2, y_2), \dots, (D_m, y_m)\}$, where $D_i \in \mathbb{R}^n$ is the input representation of the i^{th} document, $y_i \in \{0, 1\}^L$ represents the set of active/annotated labels for it ($y_i^l = 1 \iff D_i$ is labelled with l), and L is the total number of labels. Importantly, the labels are assumed to be nodes of an unknown, yet fixed, hierarchy. Using this prior knowledge and the training data, the goal is to learn a classifier that generalises well for labelling new documents.

Classical text classification methods ignore the informative prior knowledge that the set of labels form a hierarchy. Most of the hierarchical multi-class classification models assume that the hierarchy over the labels is completely known, which might not be a pragmatic assumption, since constructing hierarchies is an expensive process, especially when the number of labels is large (Bhattachia et al., 2016). In contrast, here we assume no explicit information regarding the hierarchy other than it’s existence, and the implicit information encoded in the training data. Also, in our set-up, we do not restrict the labels to be the leaves nodes in the hierarchy. As motivated earlier, here we propose to learn a classifier that jointly learns the classifier parameters as well as the label embeddings.

4.2 Our Model: HIDDEN

Our proposed model HIDDEN has two key components: one for representing the documents that may lead to well-generalizing classifiers and the other for embedding the labels in a hyperbolic space. Recall that hyperbolic spaces have shown to be well-suited for data satisfying hierarchical relations.

Document Model \mathcal{F}_w accepts as input a document, D , and outputs a n -dimensional representation of it, $\mathcal{F}_w(D) \in \mathbb{R}^n$. Here, w is the set of parameters to be learnt. In this work, we use TextCNN (Kim, 2014) as the document model. But our approach remains valid irrespective of the chosen document model.

Label Embedding Model \mathcal{G}_Θ accepts as input a label l and outputs a finite dimensional representation, $\mathcal{G}_\Theta(l)$. Here, Θ is the set of parameters to be learnt. In this work, following Nickel and Kiela (2018), we employ the simple look-up based model defined by $\mathcal{G}_\Theta(l) \equiv \Theta * y^l = \Theta_l$, where $\Theta \in \mathbb{R}^{n \times L}$ and Θ_l is the l^{th} column of Θ . These Euclidean embeddings Θ_l are then projected onto the Poincare manifold using the transformation $\Pi(x) = \frac{x}{1 + \sqrt{1 + \|x\|_2^2}}$. In summary, the hyperbolic embedding of label, l , is given by $\Pi(\Theta_l)$.

We next assume that there exists some optimal set of parameters w^* , Θ^* such that the labels annotated/active for a document, D , are exactly those whose label representations are highly aligned with that of D 's representation. Here, alignment between the representations is intended to model the natural intuition of appropriateness between label and document. Following the principle of large-margin separation, in this paper we employ the **alignment model** defined below:

$$\hat{y}_D^l(w, \Theta) \equiv \sigma \left(\mathcal{F}_w(D)^\top \Theta_l \right) \quad (2)$$

where $\hat{y}_D^l(w, \Theta)$ denotes the alignment between the document, D , and the l^{th} label as per the model with parameters (w, Θ) , and σ is the Sigmoid activation function.

Inference: Given the learnt parameters $(\hat{w}, \hat{\Theta})$, the labels with $\hat{y}_D^l(\hat{w}, \hat{\Theta}) > 0.5$ are predicted to be the active ones for D . We next detail the proposed joint objective for learning the parameters.

4.3 Joint Objective

The proposed objective consists of two terms: the first is an empirical multi-label loss term over the training data, and the second is a loss for ensuring that the hyperbolic label embeddings respect

the pairwise label co-occurrence or any other such (pairwise) partial information regarding the underlying label hierarchy.

First Term is simply a binary cross entropy loss to promote high alignment scores for each annotated label and vice-versa:

$$\mathcal{L}_1(w, \Theta) = \sum_{i=1}^m \sum_{l=1}^L \left[y_i^l \log \left(\hat{y}_i^l(w, \Theta) \right) + (1 - y_i^l) \log \left(1 - \hat{y}_i^l(w, \Theta) \right) \right] \quad (3)$$

where \hat{y}_i^l is a short-hand for $\hat{y}_{D_i}^l$.

Second Term induces lesser geodesic distance in the hyperbolic space between the label embeddings that have higher co-occurrences than those between label pairs that have less co-occurrence (Nickel and Kiela, 2018):

$$\mathcal{L}_2(\Theta) = \sum_{\substack{l, l' \in L, \\ l' \neq l}} \log \left(\frac{e^{-d(\Pi(\Theta_l), \Pi(\Theta_{l'}))}}{\sum_{z \in \mathcal{N}(l, l')} e^{-d(\Pi(\Theta_l), \Pi(\Theta_{l'}))}} \right) \quad (4)$$

where d is the metric in the hyperbolic space given by Eq.1, $\Pi(\Theta_l)$ is hyperbolic embedding of l^{th} label, and $\mathcal{N}(l, l')$ is the set of all labels that less frequently co-occur with l than l' co-occurs with l .

The overall objective function is a weighted sum of the two components described above

$$\mathcal{L}(w, \Theta) = \mathcal{L}_1(w, \Theta) + \lambda \mathcal{L}_2(\Theta) \quad (5)$$

We refer to the model corresponding to the parameters $(w_{\text{jnt}}, \Theta_{\text{jnt}})$ that minimizes this joint objective in Eq.5 as **HIDDEN_{jnt}**:

$$(w_{\text{jnt}}, \Theta_{\text{jnt}}) \in \arg \min_{w, \Theta} \mathcal{L}(w, \Theta) \quad (6)$$

Both components of our loss interact with each other to minimize the distance between document and label embeddings in the hyperbolic space. The advantage of the joint learning is well illustrated when HIDDEN_{jnt} is compared with the following baseline, henceforth referred to as **HIDDEN_{cas}**:

- (1) \mathcal{L}_2 is minimized to obtain label embeddings $\hat{\Theta}_{\text{cas}} \in \arg \min_{\Theta} \mathcal{L}_2(\Theta)$.
- (2) These are then used in \mathcal{L}_1 to obtain document parameters: $\hat{w}_{\text{cas}} \in \arg \min_w \mathcal{L}_1(w, \hat{\Theta}_{\text{cas}})$.

We also empirically compare with the following multi-class classification baseline, henceforth referred to as **HIDDEN_{flt}**:

- (1) Θ_{flt} is fixed to the identity matrix.

(2) These are then used in \mathcal{L}_1 to obtain document parameters: $\hat{w}_{\text{flat}} \in \arg \min_w \mathcal{L}_1(w, \hat{\Theta}_{\text{flat}})$.

To evaluate the benefit of using hyperbolic spaces for embedding labels, we also compare with a variant of $\text{HIDDEN}_{\text{jnt}}$ called $\text{HIDDEN}_{\text{Euc}}$ for which \mathcal{L}_2 is modified to be

$$\mathcal{L}_{2\text{Euc}}(\Theta) = \sum_{\substack{l, l' \in L, \\ l' \neq l}} \log \left(\frac{e^{-\|\Theta_l - \Theta_{l'}\|_2}}{\sum_{z \in \mathcal{N}(l, l')} e^{-\|\Theta_l - \Theta_{l'}\|_2}} \right) \quad (7)$$

to obtain the parameters $(w_{\text{Euc}}, \Theta_{\text{Euc}})$ as $(w_{\text{Euc}}, \Theta_{\text{Euc}}) \in \arg \min_{w, \Theta} \mathcal{L}_1(w, \Theta) + \lambda \mathcal{L}_{2\text{Euc}}(\Theta)$

Note that none of the variants of HIDDEN assume any explicit information regarding the underlying hierarchy. However, the former three exploit the prior knowledge that there exists a label hierarchy; whereas the latter, which is the classical multi-label classification network, completely ignores this useful information. Moreover, since the proposed model $\text{HIDDEN}_{\text{jnt}}$ performs joint learning, it is expected that $\text{HIDDEN}_{\text{jnt}}$ not only achieves better generalization, but also leads to better label embeddings, when compared to $\text{HIDDEN}_{\text{cas}}$. The simulation results in section 5 confirm the same.

4.4 Training Details

In all our experiments, the initial word embedding layer of TextCNN in the document model is initialized using 300 dimensional GloVe embeddings (Pennington et al., 2014). Following Nickel and Kiela (2017), we randomly initialize Θ from the uniform distribution $\mathcal{U}(-0.001, 0.001)$. Both the document and label representations are of length $n = 300$. We randomly choose 10% of training set as the validation set and report test set results on the best validation epoch. During training, dropout is applied to the outputs of document model as well the label model with probabilities 0.1 and 0.6 respectively. We found $\lambda = 0.1$ to yield the best validation performance. The number of training epochs are set to 30 for all experiments. Both models are optimized using stochastic gradient descent using Adam optimizer (Kingma and Ba, 2014) with learning rate as 0.001 for TextCNN.

We run all our experiments on Nvidia RTX 2080 Ti GPUs 12 GB RAM over Intel Xeon Gold 5120 CPU having 56 cores and 256 GB RAM. It takes around 1, 2 and 5 hours to train the model on RCV1, NYT and Yelp datasets respectively.

5 Experiments

In this section, we compare our approach against the baseline models and other state-of-the-art HMC approaches. First we describe the evaluation metrics and illustrating results in a synthetic setting.

5.1 Evaluation Measures

Classifier Evaluation Measures: We use standard measures for evaluating any HMC, viz., **Macro-F1** and **Micro-F1**. Let TP , TN , FN , FP denote the true positive, true negative, false negative and false positive labels respectively. Precision is $\frac{TP}{TP+FP}$ and recall is $\frac{TP}{TP+FN}$. F1-score is the harmonic mean of precision and recall. **Macro-F1** assigns equal weightage to each class and is computed as the averaged F1 score over all classes. **Micro-F1** is the F1-score computed over all instances.

Label Embedding Evaluation Measures: For a given application, let us say \mathcal{H}_* is provided to us as the ground truth hierarchy of labels/nodes, which was assumed to be unknown in our problem formulation in Section 4.1. Recall that none of the variants of HIDDEN has access to \mathcal{H}_* . How consistent with respect to \mathcal{H}_* are the label embeddings learnt by these models? We attempt to assess this consistency by adopting standard measures such as Spearman’s rank correlation coefficient (Zar, 2005) and Normalized Discounted cumulative gain (NDCG) (Järvelin and Kekäläinen, 2002).

Recall from Section 4.3, that the hyperbolic embedding for label l is $\Pi(\Theta_l)$ and likewise for l' , it is $\Pi(\Theta_{l'})$. The model parameters Θ might be learnt using any variant of HIDDEN . Given a query label, l , the geodesic distance $d(\Pi(\Theta_l), \Pi(\Theta_{l'}))$ is used to rank all other labels $l' \neq l$; smaller the distance, larger the rank. Any two labels $l' \neq l''$ that are at the same geodesic distance from l , will be assigned the same rank $r = r''$. Next, we define a graded relevance score for labels with respect to the ground truth hierarchy, \mathcal{H}_* . For any given query label $l \in \mathcal{H}_*$, we also assign a graded relevance $rel' \in \mathbb{N}$ to every other label $l' \neq l$ based on the distance (number of hops $\text{hops}(l, l')$) of l' from l in the hierarchy $\mathcal{H}_{\mathcal{D}}$; smaller the distance, larger the graded relevance (we considered $rel' \propto \frac{1}{\text{hops}(l, l')}$, for example).

Discounted Cumulative Gain (DCG) (Järvelin and Kekäläinen, 2002) is a standard measure of the quality of ranking of an approach with respect to the graded relevance provided in the ground truth. $\text{DCG}@k$ measures items (eg: labels l') k hops away

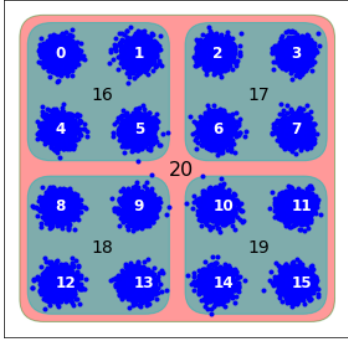


Figure 1: Gaussian used for the synthetic experiment

from the query l . The gain is accumulated from the top of the ranked list upto some pre-specified position k in the list, with the gain of each result discounted at lower ranks: $DCCG_k = \sum_{i=1}^k \frac{rel_i}{\log_2(i+1)}$. Here, rel_i is the graded relevance at position i . This result is itself averaged over all query labels $l \in \mathcal{H}_*$.

Spearman's rank correlation coefficient denoted by r is a non-parametric metric to measure statistical dependence between ranking of two variables. For each query label (l), we first measure rank correlation between the predicted rank r_p of a label l' and its own rank r_h as per ground truth hierarchy h . The correlation coefficient between r_{l_p} and r_{l_h} is computed as $r_l = \frac{cov(r_p, r_h)}{\sigma_{r_p} \sigma_{r_h}}$. Here, cov is the covariance and σ , the standard deviation. The final score, r is the averaged score across all labels l' in the set.

5.2 Validation via Synthetic Experiments

To observe the behaviour of the proposed approach HIDDEN with respect to the evaluation measures in a controlled environment, we present one such synthetic setup. The goal in this section is to illustrate the advantage of joint learning of parameters over isolated learning. Consider 2D data generated from 16 neatly separated Gaussians laid out on a grid as illustrated in Figure 1. Each of the 16 gaussians corresponds to a single label $l_1, l_2 \dots l_{16}$. We consider a second layer of 4 labels $l_{17} \dots l_{20}$, obtained by grouping the gaussians into 4; each quadrant in the larger square would correspond to a single label. Finally, we have a third layer consisting of a single label l_{21} - viz., the entire large square. This simple hierarchy is hidden from our variants of HIDDEN as well as from the flat model. The synthesized data is split randomly into train-test in the ratio 60:40. For each of the jointly optimised model HIDDEN_{jnt}, the cas-

Table 1: Comparison of methods on the Synthetic **SETTING 1** with respect to increasing probability with which a label is randomly dropped. Synthetic data used here has 12000 training and 8000 test samples.

Prob	0.00		0.20		0.40	
	Micro F1	Macro F1	Micro F1	Macro F1	Micro F1	Macro F1
HIDDEN _{fit}	96.8	89.1	93.2	87.8	90.4	87.7
HIDDEN _{cas}	98.0	93.4	94.4	88.9	91.9	91.0
HIDDEN _{jnt}	98.1	94.0	94.8	91.6	92.3	91.7

caded model HIDDEN_{cas} as well as the flat model, we observe (i) the performance of the classification models $\mathcal{F}_w(D)$ measured in terms of **Micro-F1** and **Macro-F1** as well as (ii) the consistency of the label embedding models $\mathcal{G}_\Theta(l)$ with respect to the hidden 3-level hierarchy over the 21 labels. We record observations in two settings:

SETTING 1 in which *for each training instance, one of the annotated labels are dropped, uniformly at random*: In Table 1 we note the performances of the different approaches with increasing rate at which labels are dropped. The jointly optimised model HIDDEN_{jnt} accounts for the classification task through loss component \mathcal{L}_1 as also the somewhat redundant label co-occurrence through the loss component \mathcal{L}_2 .

As expected, we observe that the performance of HIDDEN_{jnt} is more robust to this form of label noise than the HIDDEN_{cas} and HIDDEN_{fit} models. This is because HIDDEN_{fit} entirely relies on the training data and ignores the prior knowledge of existence of a label hierarchy. HIDDEN_{cas} is also less robust as it over-relies on the label co-occurrence by minimising \mathcal{L}_2 (which, in isolation will be sensitive to label noise), before venturing into the classification task by minimising \mathcal{L}_1 .

SETTING 2 in which the size of the *training set is decreased without corrupting labels*: We observe the performances of the different approaches with decreasing size of the training set and note that the performance of the jointly optimised model HIDDEN_{jnt} falls back on the label correlation signals through the loss component \mathcal{L}_2 and is therefore more robust to decreasing size of the data set than the flat classifier. We observed similar results for other synthetic settings. Owing to space constraints, the plots and other ranking results are provided in the supplementary material.

5.3 Real-world Text Datasets

We used three datasets, namely, RCV1, Yelp and NYT in our experiments:

(1) **RCV1** (Lewis et al., 2004) - RCV1 is a

newswire dataset of the articles collected between 1996-1997 from Reuters

- (2) **NYT** (Sandhaus, 2008) - This corpus contains articles from New York Times published between January 1st, 1987 and June 19th, 2007
- (3) **Yelp**¹ - Yelp is a review dataset of restaurants and each review is labelled with hierarchical categories of restaurants. Following the experimental design in Mao et al. (2019), we use the set of reviews for a business to predict the categories to which the business belongs.

Some statistics pertaining to these datasets are presented in Table 2.

Table 2: Statistics of the datasets used in the experiments. $|L|$ denotes the number of labels, $\text{Avg}(|L|)$ is the average number of labels per instance and $\text{Max}(|L|)$ is the maximum number of labels for an instance.

Dataset	Hierarchy	$ L $	$\text{Avg}(L)$	$\text{Max}(L)$	Train	Val	Test
RCV1	Tree	104	3.24	17	20833	2314	781265
NYT	Tree	120	6.58	24	86461	9606	9903
Yelp	DAG	539	4.07	32	98460	10939	46884

5.4 Comparison of models that do not use the true hierarchy

We compare performance of the different models that do not use the true hierarchy. These include our flat baseline $\text{HIDDEN}_{\text{flt}}$, the cascaded model $\text{HIDDEN}_{\text{cas}}$ as well as our joint model $\text{HIDDEN}_{\text{jnt}}$. We compare them against the baseline TextCNN-flat model reported in Mao et al. (2019). The results are presented in Table 3 for $\lambda = 0.1$.

Overall, our baseline ($\text{HIDDEN}_{\text{flt}}$) performs better than the previous baseline with the exception of Macro-F1 on NYT. We observe improvement of the joint model $\text{HIDDEN}_{\text{jnt}}$ over the flat ($\text{HIDDEN}_{\text{flt}}$) and cascaded ($\text{HIDDEN}_{\text{cas}}$) models on RCV1 and NYT (for each of which, the labels form a tree) in Table 3. However, on the Yelp dataset, the cascaded model ($\text{HIDDEN}_{\text{cas}}$) performs somewhat worse (-2 Micro-F1 and -3.3 Macro-F1) than our baseline model ($\text{HIDDEN}_{\text{flt}}$), hinting at the possibility that label co-occurrence information might not be helpful toward the classification task. This could be partly also because the labels in Yelp are structured in the form of a DAG. Constant curvature property of hyperbolic spaces makes them unsuitable for learning DAG structures (Li et al., 2018). However, the Macro-F1 performance of $\text{HIDDEN}_{\text{jnt}}$ is far better than that of the cascaded model $\text{HIDDEN}_{\text{cas}}$.

¹<https://www.yelp.com/dataset/challenge>

This illustrates that our joint model is able to better recover from less reliable (or less useful) label co-occurrence information, just as was illustrated in the Table 1 for the synthetic setting.

Table 3: Performance comparison on all three datasets with TextCNN as the base classification model. Recall that $\text{HIDDEN}_{\text{flt}}$ is our own multi-class classification baseline. We observe that the numbers reported by Mao et al. (2019) (indicated by *) for the TextCNN based flat baseline model are consistently outperformed by the proposed HIDDEN models.

Dataset	Method	Micro-F1	Macro-F1
RCV1	TextCNN-Flat*	76.6	43.0
	$\text{HIDDEN}_{\text{flt}}$	77.9	44.5
	$\text{HIDDEN}_{\text{cas}}$	78.0	45.5
	$\text{HIDDEN}_{\text{jnt}}$	79.3	47.3
NYTimes	TextCNN-Flat*	69.5	39.5
	$\text{HIDDEN}_{\text{flt}}$	76.4	37.1
	$\text{HIDDEN}_{\text{cas}}$	74.6	33.2
	$\text{HIDDEN}_{\text{jnt}}$	77.0	43.6
Yelp	TextCNN-Flat*	62.8	27.3
	$\text{HIDDEN}_{\text{flt}}$	62.5	37.9
	$\text{HIDDEN}_{\text{cas}}$	60.5	33.9
	$\text{HIDDEN}_{\text{jnt}}$	60.8	35.6

5.5 Comparison of Hyperbolic space and Euclidean space

To assess the utility of the hyperbolic space for embedding hierarchical labels, we compare $\text{HIDDEN}_{\text{jnt}}$ and $\text{HIDDEN}_{\text{euc}}$. Table 4 presents this comparison on the three datasets. $\text{HIDDEN}_{\text{euc}}$ performs worse than $\text{HIDDEN}_{\text{jnt}}$ which uses the hyperbolic space for embedding labels (except Micro-F1 for Yelp due to reasons stated before). This is expected since embedding trees is much more effective in the hyperbolic space compared to Euclidean space since in the hyperbolic space, volume grows exponentially with distance from the origin while in Euclidean space, this growth is polynomial. The number of nodes in a tree also increases exponentially with distance from the root, making Hyperbolic spaces useful for embedding hierarchies.

5.6 Comparison with model that explicitly uses the true hierarchy

We compare performance of our joint approach $\text{HIDDEN}_{\text{jnt}}$ against a state-of-the-art hierarchical multi-label classifier, HiLAP (Mao et al., 2019). However, unlike our proposed models (variants of HIDDEN), HiLAP has access to the true hierarchy

Table 4: Performance comparison for $\text{HIDDEN}_{\text{jnt}}$ with $\text{HIDDEN}_{\text{euc}}$. $\text{HIDDEN}_{\text{jnt}}$ consistently has better Macro-F1 better than $\text{HIDDEN}_{\text{euc}}$ and generally better Micro-F1 too. This illustrates the utility of Hyperbolic spaces for embedding label hierarchies

Dataset	Method	Micro-F1	Macro-F1
RCV1	$\text{HIDDEN}_{\text{euc}}$	78.4	47.6
	$\text{HIDDEN}_{\text{jnt}}$	79.3	47.3
NYTimes	$\text{HIDDEN}_{\text{euc}}$	76.4	40.4
	$\text{HIDDEN}_{\text{jnt}}$	77.0	43.6
Yelp	$\text{HIDDEN}_{\text{euc}}$	61.1	34.2
	$\text{HIDDEN}_{\text{jnt}}$	60.8	35.6

both training and inference. Thus, HiLAP serves as some form of skyline for the HIDDEN suite of approaches proposed in this paper. HiLAP learns label assignment policy using the reinforcement learning framework.

In Table 5, we compare the performance of $\text{HIDDEN}_{\text{jnt}}$ against HiLAP model as reported in Mao et al. (2019). Interestingly, on RCV1, we obtain better Micro-F1 score (+0.7) for the joint model $\text{HIDDEN}_{\text{jnt}}$ over the HiLAP method. On NYT, our Micro-F1 score is far better (+7.1) than HiLAP; our Macro-F1 score is also marginally better (+0.4) than their Macro-F1 scores. These results are interesting because $\text{HIDDEN}_{\text{jnt}}$ seems to obtain better generalisation through joint learning of the document classifier and label embeddings in a hyperbolic space, even without access to the true hierarchy. However, on Yelp, HiLAP seems to benefit over $\text{HIDDEN}_{\text{jnt}}$ by explicitly using the true hierarchy.

Table 5: Performance comparison of $\text{HIDDEN}_{\text{jnt}}$ with HiLAP with respect to Macro-F1 and Micro-F1. It is interesting to note here that though $\text{HIDDEN}_{\text{jnt}}$ does not know the true hierarchy, it performs better than HiLAP (which uses the true hierarchy) in some cases. (HiLAP numbers are those reported by Mao et al. (2019))

Dataset	$\text{HIDDEN}_{\text{jnt}}$		HiLAP	
	Micro	Macro	Micro	Macro
RCV1	79.3	47.3	78.6	50.5
NYTimes	77.0	43.6	69.9	43.2
Yelp	60.8	35.6	65.5	37.3

5.7 Evaluating performance of embeddings

We compare embeddings learned using different approaches with the ground truth hierarchy to eval-

uate the effectiveness of the embeddings. Figure 2 shows the plot of NDCG scores for different values of k on the RCV1 and NYTimes dataset across $\text{HIDDEN}_{\text{cas}}$ and $\text{HIDDEN}_{\text{jnt}}$ (for two different values of λ). In Table 6, we compare the Spearman rank correlation. The superior performance of $\text{HIDDEN}_{\text{jnt}}$ strongly suggests that the embeddings learnt using the joint model are more representative of the true hierarchical organisation of the labels than those obtained using the flat and cascaded variants. This also goes to show that even the first term in our objective has positive contribution towards the learning of hyperbolic embeddings and indeed joint learning is beneficial.

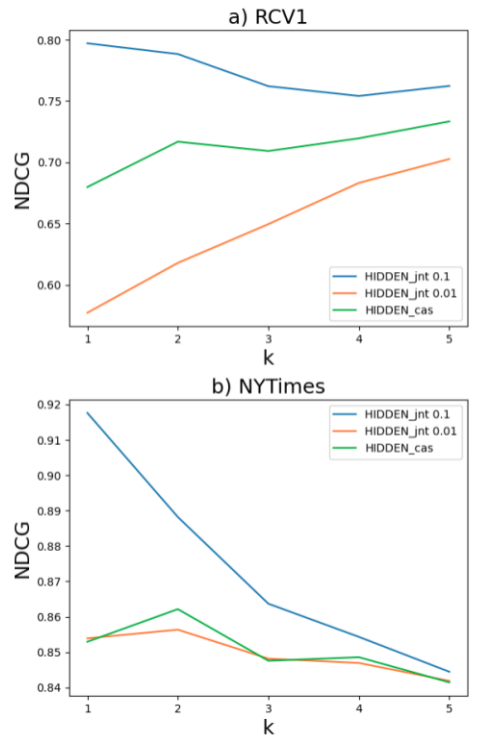


Figure 2: Plot of NDCG versus k for assessing the quality of the learnt label embeddings with respect to the actual hierarchy on RCV1 and NYT datasets. The better performance of $\text{HIDDEN}_{\text{jnt}}$ indicates the label embeddings Θ_{jnt} are most representative of the true hierarchy.

6 Conclusion

We propose a novel approach to hierarchical multi-label classification based on joint learning of document classifier and label embeddings in hyperbolic space. The proposed framework HIDDEN allows us to discover label hierarchical relationship by leveraging properties of hyperbolic geometry. Even though label-hierarchy is assumed to be unavailable, our method achieves comparable results

Table 6: Spearman rank correlation test for the generated embeddings for all the datasets. Each method is compared against the ground truth hierarchy. Highest values for $\text{HIDDEN}_{\text{jnt}}$ indicates the label embeddings Θ_{jnt} are most representative of the true hierarchy.

	$\text{HIDDEN}_{\text{fit}}$	$\text{HIDDEN}_{\text{jnt}}$	$\text{HIDDEN}_{\text{cas}}$
RCV1	21.2	53.9	44.1
NYTimes	11.4	39.5	36.1
Yelp	16.3	31.9	28.8

with state-of-the-art *hierarchy aware* methods. We performed extensive experiments on three datasets and demonstrate effectiveness of the learned embeddings.

Acknowledgements

We thank anonymous reviewers for providing constructive feedback. Ayush Maheshwari is supported by a Fellowship from Ekal Foundation (www.ekal.org). We are also grateful to IBM Research, India (specifically the IBM AI Horizon Networks - IIT Bombay initiative) for their support and sponsorship.

References

Rahul Agrawal, Archit Gupta, Yashoteja Prabhu, and Manik Varma. 2013. Multi-label learning with millions of labels: Recommending advertiser bid phrases for web pages. In *Proceedings of the 22nd international conference on World Wide Web*, pages 13–24.

Ramakrishna Bairi, Rishabh K. Iyer, Ganesh Ramakrishnan, and Jeff A. Bilmes. 2015. [Summarization of multi-document topic hierarchies using submodular mixtures](#). In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics (ACL 2015)*, pages 553–563.

Ramakrishna B. Bairi, Mark James Carman, and Ganesh Ramakrishnan. 2016. [Beyond clustering: Sub-dag discovery for categorising documents](#). In *Proceedings of the 25th ACM International Conference on Information and Knowledge Management (CIKM 2016)*, 2016, pages 801–810.

Kush Bhatia, Kunal Dahiya, Himanshu Jain, Anshul Mittal, Yashoteja Prabhu, and Manik Varma. 2016. [The extreme classification repository: Multi-label datasets and code](#).

Ricardo Cerri, Rodrigo C Barros, and André CPLF de Carvalho. 2011. Hierarchical multi-label classification for protein function prediction: A local approach based on neural networks. In *2011 11th Inter-*

national Conference on Intelligent Systems Design and Applications, pages 337–343. IEEE.

Nicolò Cesa-Bianchi, Claudio Gentile, and Luca Zani-boni. 2006. Hierarchical classification: combining bayes with svm. In *Proceedings of the 23rd international conference on Machine learning*, pages 177–184.

Boli Chen, Xin Huang, Lin Xiao, Zixin Cai, and Lip-ing Jing. 2019. Hyperbolic interaction model for hierarchical multi-label classification. *arXiv preprint arXiv:1905.10802*.

Siddharth Gopal and Yiming Yang. 2013. Recursive regularization for large-scale classification with hierarchical and graphical dependencies. In *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 257–265.

Mikhael Gromov. 1987. Hyperbolic groups. In *Essays in group theory*, pages 75–263. Springer.

Kalervo Järvelin and Jaana Kekäläinen. 2002. Cumulated gain-based evaluation of ir techniques. *ACM Transactions on Information Systems (TOIS)*, 20(4):422–446.

Rie Johnson and Tong Zhang. 2015. Effective use of word order for text categorization with convolutional neural networks. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 103–112.

Yoon Kim. 2014. Convolutional neural networks for sentence classification. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1746–1751.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Daphne Koller and Mehran Sahami. 1997. Hierarchically classifying documents using very few words. Technical report, Stanford InfoLab.

Dmitri Krioukov, Fragkiskos Papadopoulos, Maksim Kitsak, Amin Vahdat, and Marián Boguná. 2010. Hyperbolic geometry of complex networks. *Physical Review E*, 82(3):036106.

Ashish Kulkarni, Narasimha Raju Uppalapati, Pankaj Singh, and Ganesh Ramakrishnan. 2018. [An interactive multi-label consensus labeling model for multi-label judgments](#). In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence (AAAI-18)*, 2018, pages 1479–1486.

David D Lewis, Yiming Yang, Tony G Rose, and Fan Li. 2004. Rcv1: A new benchmark collection for text categorization research. *Journal of machine learning research*, 5(Apr):361–397.

- Xiang Li, Luke Vilnis, Dongxu Zhang, Michael Boratko, and Andrew McCallum. 2018. Smoothing the geometry of probabilistic box embeddings. In *International Conference on Learning Representations*.
- Jingzhou Liu, Wei-Cheng Chang, Yuexin Wu, and Yiming Yang. 2017. Deep learning for extreme multi-label text classification. In *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 115–124.
- Yuning Mao, Jingjing Tian, Jiawei Han, and Xiang Ren. 2019. Hierarchical text classification with reinforced label assignment. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 445–455.
- Andrew Kachites McCallum. 1999. Multi-label text classification with a mixture model trained by em. In *AAAI 99 workshop on text learning*. Citeseer.
- Maximillian Nickel and Douwe Kiela. 2017. Poincaré embeddings for learning hierarchical representations. In *Advances in neural information processing systems*, pages 6338–6347.
- Maximillian Nickel and Douwe Kiela. 2018. Learning continuous hierarchies in the lorentz model of hyperbolic geometry. In *International Conference on Machine Learning*, pages 3779–3788.
- Hao Peng, Jianxin Li, Yu He, Yaopeng Liu, Mengjiao Bao, Lihong Wang, Yangqiu Song, and Qiang Yang. 2018. Large-scale hierarchical text classification with recursively regularized deep graph-cnn. In *Proceedings of the 2018 World Wide Web Conference*, pages 1063–1072.
- Shengwen Peng, Ronghui You, Hongning Wang, Chengxiang Zhai, Hiroshi Mamitsuka, and Shan-feng Zhu. 2016. Deepmesh: deep semantic representation for improving large-scale mesh indexing. *Bioinformatics*, 32(12):i70–i79.
- Jeffrey Pennington, Richard Socher, and Christopher D Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543.
- Juho Rousu, Craig Saunders, Sandor Szedmak, and John Shawe-Taylor. 2006. Kernel-based learning of hierarchical multilabel classification models. *Journal of Machine Learning Research*, 7(Jul):1601–1626.
- Evan Sandhaus. 2008. The new york times annotated corpus. *Linguistic Data Consortium, Philadelphia*, 6(12):e26752.
- Carlos N Silla and Alex A Freitas. 2011. A survey of hierarchical classification across different application domains. *Data Mining and Knowledge Discovery*, 22(1-2):31–72.
- Carlos N Silla Jr and Alex A Freitas. 2009. A global-model naive bayes approach to the hierarchical prediction of protein functions. In *2009 Ninth IEEE International Conference on Data Mining*, pages 992–997. IEEE.
- Isaac Triguero and Celine Vens. 2016. Labelling strategies for hierarchical multi-label classification techniques. *Pattern Recognition*, 56:170–183.
- Ke Wang, Senqiang Zhou, and Yu He. 2001. Hierarchical classification of real life documents. In *Proceedings of the 2001 SIAM International Conference on Data Mining*, pages 1–16. SIAM.
- Jonatas Wehrmann, Ricardo Cerri, and Rodrigo Barros. 2018. Hierarchical multi-label classification networks. In *International Conference on Machine Learning*, pages 5075–5084.
- Bishan Yang, Jian-Tao Sun, Tengjiao Wang, and Zheng Chen. 2009. Effective multi-label active learning for text classification. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 917–926.
- Jerrold H Zar. 2005. Spearman rank correlation. *Encyclopedia of Biostatistics*, 7.

Appendix

1 Explanation for $\Pi(x)$

The Lorentz model is defined as the Riemannian Manifold, $\mathcal{L}^n = (\mathcal{H}^n, g_l)$, where $\mathcal{H}^n = \{\mathbf{x} \in \mathbb{R}^{n+1} : \langle \mathbf{x}, \mathbf{x} \rangle_{\mathcal{L}} = -1, \mathbf{x}_0 > 0\}$, and $g_l = \text{diag}([-1 \ 1 \ \dots \ 1])$. Here, $\langle \mathbf{x}, \mathbf{y} \rangle_{\mathcal{L}}$, known as the Minkowski inner-product, is given by

$$\langle \mathbf{x}, \mathbf{y} \rangle_{\mathcal{L}} = -x_0y_0 + x_1y_1 + \dots + x_ny_n$$

The Poincaré model and the Lorentz model are equivalent in isometry. Therefore points in Lorentz manifold can be mapped into Poincaré ball as, $p : \mathcal{H}^n \rightarrow \mathcal{P}^n$

$$p(x_0, x_1, \dots, x_n) = \frac{(x_1, \dots, x_n)}{x_0 + 1}$$

A point \mathbf{x} in the Euclidean space \mathbb{R}^n can be projected onto the Lorentz manifold \mathcal{H}^n using the transformation $\Omega(\mathbf{x}) = \left[\sqrt{1 + \|\mathbf{x}\|^2}, \mathbf{x} \right]$. This transformation ensures that the Minkowski inner-product, $\langle \Omega(\mathbf{x}), \Omega(\mathbf{x}) \rangle_{\mathcal{L}}$, is equal to -1 , and that the first component, $\Omega(\mathbf{x})_0 \equiv \sqrt{1 + \|\mathbf{x}\|^2}$ is positive, as required for membership in the Lorentz manifold.

Now using the isometry between Poincaré and Lorentz models (Nickel and Kiela, 2018), we have $\Pi : \mathbb{R}^n \rightarrow \mathcal{P}^n$ as

$$\Pi(\mathbf{x}) = p(\Omega(\mathbf{x})) = \frac{\mathbf{x}}{1 + \sqrt{1 + \|\mathbf{x}\|^2}}$$

2 Dataset Details

We describe the details of the datasets used in our experiments.

For RCV1 dataset (Lewis et al., 2004), we use the original training/test split and use 10% of the training set as the validation set. We introduce an extra *Root* label in addition to the 103 labels present in the dataset. Each document in the dataset is labelled with this label.

The details for the other datasets used are same as in Mao et al. (2019) and we refer the readers to the same.

3 Remarks on Synthetic Experiments

In **SETTING 1**, with increasing probability of a label being randomly dropped, we observe in Figure 3 that the performance of all the models decreases which is expected. However, it is interesting to note that $\text{HIDDEN}_{\text{jnt}}$ is more robust to

the noisy labels and always performs better than $\text{HIDDEN}_{\text{flt}}$ since it has an additional source of information about the labels via label co-occurrences. This information is also implicitly available to $\text{HIDDEN}_{\text{flt}}$ but from our experiments, we observe that providing this information explicitly improves performance. We also observe that performance of $\text{HIDDEN}_{\text{cas}}$ fluctuates quite a bit which is probably due to the fact that it overly relies on label co-occurrence and commits to a set of label embedding before trying to solve the classification task.

In **SETTING 2**, with increasing number of points in the training set, we observed in Figure 4 that with very small datasets, $\text{HIDDEN}_{\text{cas}}$ performs slightly better than $\text{HIDDEN}_{\text{jnt}}$ but with increasing dataset sizes, $\text{HIDDEN}_{\text{jnt}}$ performs better. Since there is no label noise in this setting, the label co-occurrences are expected to be quite meaningful and with small dataset sizes, jointly learning both the document embedding model \mathcal{F}_w and the label embeddings Θ jointly ($\text{HIDDEN}_{\text{jnt}}$) is more difficult compared learning Θ first and then \mathcal{F}_w ($\text{HIDDEN}_{\text{cas}}$).

$\text{HIDDEN}_{\text{jnt}}$ and $\text{HIDDEN}_{\text{cas}}$ perform better than $\text{HIDDEN}_{\text{flt}}$ since the former can fall back of the label correlation signal via \mathcal{L}_2 which is available in a much more easily usable format than to $\text{HIDDEN}_{\text{flt}}$. As discussed above, having the label co-occurrences available explicitly to the models help achieve better performance.

We observe similar results in other synthetic settings.

4 Hierarchy of the Synthetic data

As described in the paper, the synthetic data is generated from 16 Bivariate Gaussian distributions with their means placed evenly in a 4×4 grid. These Gaussians are then grouped at various levels to get an hierarchy with 3 level as shown in Figure 5.

5 Contrasting label hierarchies across datasets

Recall, how in Table 3 of the paper, we observe improvement of the joint model $\text{HIDDEN}_{\text{jnt}}$ over the flat ($\text{HIDDEN}_{\text{flt}}$) and cascaded ($\text{HIDDEN}_{\text{cas}}$) models on RCV1 and NYTimes datasets. Note that the labels of RCV1 as well as of NYTimes form trees. However, on the Yelp dataset, the cascaded model ($\text{HIDDEN}_{\text{cas}}$) performs somewhat worse (-2 on micro and -3.3 on macro) than our baseline

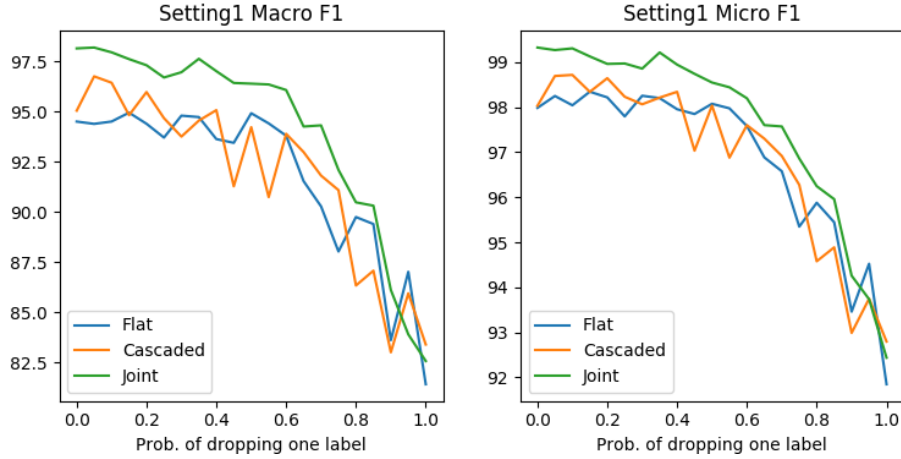


Figure 3: Micro and Macro F1 scores with increasing probability of randomly dropping one label for all three model. Scores are averaged over 10 runs

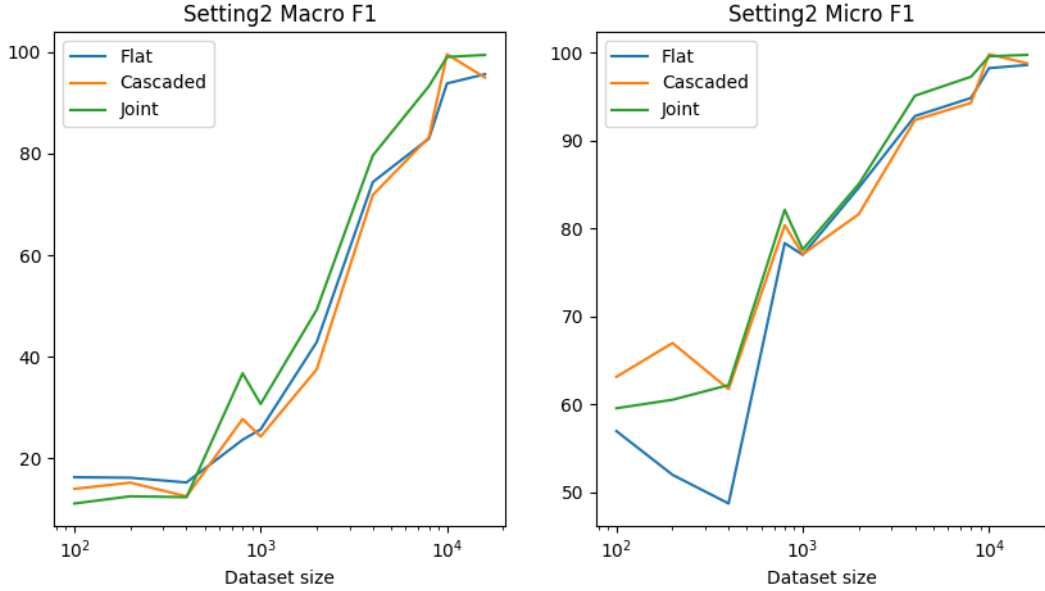


Figure 4: F1 scores with increasing dataset sizes (number of points in train set) for all three model

model ($HIDDEN_{ft}$), hinting at the possibility that label co-occurrence information might not be helpful toward the classification task. This could be partly also because the labels in Yelp are structured in the form of a DAG with 12 labels in the labels set having more than one parent.

5.1 δ -hyperbolicity

To further investigate why our method fails to perform well on Yelp dataset, we compute hyperbolicity (Gromov, 1987) for each of the label hierarchies. The hyperbolicity δ of a graph G is a measure of how tree-like the graph is. Lower the δ , the more tree-like is the graph. Hyperbolicity δ is 0 for trees.

As shown in Table 7, RCV1 has 0 hyperbolic-

Dataset	RCV1	NYT	Yelp
Hyperbolicity	0	1	1

Table 7: Hyperbolicity (Gromov, 1987) of the label hierarchies for the datasets used

ity as expected, since the label hierarchy is a tree. We would have expected NYT to also have a hyperbolicity of 0 but the label `Others` appears at different levels of the hierarchy making it a DAG. However, this is the only deviation from being a tree and thus our method is able to perform well on NYT. For Yelp, the hyperbolicity is 1 and there are multiple labels with more than one parent and thus it is less tree-like than the NYT hierarchy. It

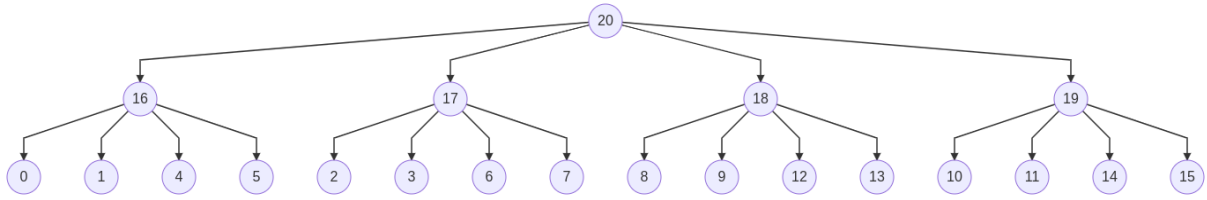


Figure 5: Hidden Hierarchy in the Synthetic experiments

is due to this DAG-ness that our method does not perform as well on Yelp as the other datasets. Note that this is a limitation of hyperbolic spaces and using other geometric spaces which are conducive to embeddings DAGs, our method is expected to perform well even when the implicit label hierarchy is a DAG. In summary, hyperbolic spaces are good for embedding low δ -hyperbolicity graphs and thus our methods perform better with graphs having low hyperbolicity.