LEGO-MM: LEarning structured model by probabilistic loGic Ontology tree for MultiMedia

Jinhui Tang, Shiyu Chang, Guo-Jun Qi, Qi Tian, Fellow, IEEE, Yong Rui, Fellow, IEEE, and Thomas S. Huang, Life Fellow, IEEE

Abstract—Recent advances in Multimedia ontology have resulted in a number of concept models, e.g., LSCOM and Mediamill 101, which are accessible and public to other researchers. However, most current research effort still focuses on building new concepts from scratch, very few work explores the appropriate method to construct new concepts upon the existing models already in the warehouse. To address this issue, we propose a new framework in this paper, termed LEGO-MM, which can seamlessly integrate both the new target training examples and the existing primitive concept models to infer the more complex concept models. LEGO-MM treats the primitive concept models as the lego toy to potentially construct an unlimited vocabulary of new concepts. Specifically, we first formulate the logic operations to be the lego connectors to combine existing concept models hierarchically in probabilistic logic ontology trees. Then, we incorporate new target training information simultaneously to efficiently disambiguate the underlying logic tree and correct the error propagation. Extensive experiments are conducted on a large vehicle domain data set from ImageNet. The results demonstrate that LEGO-MM has significantly superior performance over existing state-of-the-art methods, which build new concept models from scratch.

Index Terms—LEGO-MM, Concept recycling, Model warehouse, Probabilistic logic ontology tree, Logical operations.

I. INTRODUCTION

Effectively modeling structured concepts has become an essential ingredient for visual recognition, retrieval and search on the Web. In the prior literature, many sophisticated models have been proposed to recognize a wide range of visual concepts from our daily life to many specific domains such as news video broadcast, surveillance videos, etc. While most researchers continue to build new models from scratch, very few work explores the appropriate method to construct new concepts upon the existing models already in the warehouse. To address this issue, we propose a new framework in this paper, termed LEGO-MM, which can seamlessly integrate both the new target training examples and the existing primitive concept models to infer the more complex concept models. LEGO-MM treats the primitive concept models as the lego toy to potentially construct a wide range of visual concepts. To address such challenges, the proposed method aims to recycle the existing semantics involved in the models. Generally, a large number of samples can provide good generalization ability. However, in many cases, it is very difficult to obtain massive training data due to the expense of manual labeling work. Moreover, computational power is another constraint to recognize a wide range of visual concepts. To address such challenges, the proposed method aims to recycle the existing semantics for the new tasks.

The inspiration of our general idea comes from the lego constructing complex toys. By analogy to the lego toys, the existing models in the warehouse can be viewed as each interlocking plastic brick in the toy. Based on them, the more complex concepts can be constructed. In this way, such a Multimedia “lego” model can provide semantic-rich building blocks to construct new concepts, rather than starting with zero knowledge. This idea brings about a new perspective to efficiently leverage a large number of existing primitive concepts for constructing potentially unlimited vocabularies of visual concepts.

To construct new concepts from existing components, we
need to explore the appropriate array of gears to connect the existing lego pieces together. Similar to the toy lego, the array of gears play an essential role of coherently connecting all the components as a whole. First, let us investigate how human perceives a new concept in the real world. Usually, children are taught to learn concrete concepts which can be directly recognized by their natural attributes, such as shape, color and materials. As growing up, they start to learn how to use logics to connect these primitive concepts into more complex concepts. For instance, “beach” is a fairly abstract concept from concepts “people”, “sand”, “boat”, “sea”, etc. As illustrated in Figure 1, “sand”, “sea”, “people” and “boat” are parts of “beach”. Therefore, “beach” can be represented by “(people AND sea)” OR “(sea AND sand)” OR “(sea AND boat)” OR “(sea AND people AND sand)” which exploits the possible combinations of parts of “beach” by a AND-OR relationship. It demonstrates that the hierarchical semantic ontology can provide a reasonable way to model concepts, in an order from the primitive concepts in the lower level to the complex ones in upper level by utilizing various logical operations. In other words, given a collection of primitive concepts, many other complex concepts can be built upon these primitive concepts by connecting them with logics.

Based on the above observations, we propose a novel LEGO-MM approach, called LEarning structured model by probabilistic ioGical Ontology tree for MultiMedia in this paper, which will construct structured concepts built upon a set of primitive models. The key contributions of this paper can be summarized as follows.

- Different from many existing concept modeling techniques, LEGO-MM integrates the logical and statistical inferences in a unified framework where the existing primitive concepts are connected into a potentially unlimited vocabulary of high-level concepts by the basic logical operations. In contrast, most existing modeling algorithms either only learn a flat correlative concept structure [10] [11], or a simple hierarchical structure without logical connections [12] [13] [14].
- An efficient statistical learning algorithm is proposed to model the complex concepts in the upper levels of hierarchy upon logically connected primitive concepts. This statistical learning approach is much more flexible, where each concept in the hierarchy can be modeled from heterogeneous feature spaces of the most suitable feature descriptors (e.g., visual features such as color and shape for scenery concepts, textual features such as TF-IDF for named-entities) or can be obtained from different semantic warehouse. This means that when we build the target model, we can select the LEGO-MM made from different “materials” and still be able to connect these heterogeneous pieces of lego together.
- LEGO-MM can simultaneously incorporate both new target training information and lego building blocks. This setup allows LEGO-MM to efficiently disambiguate the underlying logic tree and correct the error propagation only leveraging a few of training samples, especially for the situation where a large amount of concepts need to be categorized, and labeled data is deficient. It has demonstrated the significant superiority compared to the SVM-type training algorithms which build new concepts from scratch.

To sum up, the general idea is that the primitive models as pieces of multimedia lego can be viewed as building blocks by analogy to training examples in conventional classification problem, because both of them provide basic semantic information to infer new concept models. In the prior work, a large number of models exist in many warehouses such as LSCOM 374 and Mediamill 101. These models can provide much more semantic resources to explore information than training examples. This is because these models are learned from example images and video key-frames, and thus rich discriminative information about the primitive concepts has been captured by them. Therefore, it is much more efficient to mine these models directly, rather than coming back to tediously collecting labeled examples and retraining models again. In this way, plenty of resources and efforts can be saved in the multimedia community by leveraging the existing multimedia lego models.

Compared to our preliminary work [15], in this paper we present more detailed theoretical derivations of the proposed approach and conduct more extensive experiments and analysis. The paper is organized as follows. We review related work in section 2. In section 3, the probabilistic logic ontology tree is formally proposed in subsection 3.1. We explain the corresponding learning algorithm for the complex concepts in probabilistic logic ontology tree in section 3.2. Then the probabilistic logic ontology tree is applied to hierarchical concept classification problem in Section 4. In section 5, we present extensive experimental results on a large vehicle domain data set from ImageNet, and demonstrate significantly superior performance over existing SVM-type approaches which build new concept models from scratch. Finally, we conclude and propose possible future directions in section 6.

II. RELATED WORK

Hierarchical concept classification has attracted much attention [16][12][17]. As opposed to the traditional flat concept classification, it attempts to classify a testing sample by a hierarchical concept tree. Hierarchical structure is a natural concept organization form which is consistent with the natural language. For example, in WordNet[18], the concepts are organized in a hierarchical structure with hyponym (i.e., Y is a hyponym of X if every Y is a kind of X) or meronym (i.e., Y is a meronym of X if Y is a part of X) relation. While the contribution discussed in [19][20][21] focussed on using hierarchical structures to enhance classification efficiency, several other works proposed to learn visual categories hierarchically [20][22][23] in an unsupervised or semi-supervised fashion. Some other researches illustrated methods for organizing low level
object representation hierarchically so that descriptiveness and discrimination performance are enhanced [24] [25].

Another important research direction is how to handle limited labeled data in visual classification and recognition. Usually, the generalization ability of most of the supervised models is determined by the number of labeled samples involved in the training stage. However, in many applications, a large number of labeled training data is hard to obtain because of the expense of labeling. A deterministic approach to alleviate this problem is to incorporate unlabeled data with model learning (co-training) using manifold approach [26].

To present how we can apply LEGO-MM to construct complex concepts, in this section we define concrete data learning structure - Probabilistic Logical Ontology Tree (PLOT).

A. Prior Probabilistic Models in PLOT

We start by the definition of PLOT with an example. As illustrated in figure 2, PLOT is a logical tree

\[
T = \{(C, f, L), (C^6, f^6, L^6), (C^7, f^7, L^7), (C^8, f^8, L^8), (C^9, f^1, P^1), \ldots, (C^5, f^5, P^5)\}
\]

(1)

where \(C\) and \(C^i\) \((i = 1, 2, \ldots, 5)\) are concept nodes, and \(f\) and \(f^i\) are different feature descriptors attached with \(C\) and \(C^i\) (it also can be seen as a sub-tree of a hierarchical ontology). For each upper level concept \(C^i\) other than leaf nodes, \(C^i\) can be expanded into a set of children concepts by a logical operation \(L^i\) from either OR, AND, or NOT. For each node, there is also an attached model \(P^i(y \mid f^i(x))\) predicting the probability of label being positive if \(y = 1\) or negative if \(y = 0\) given the feature \(f^i(x)\) for each sample \(x\). The associated model can have arbitrary flexible mathematical forms such as logistic regression model, exponential model or even support vector machine or boosting model (but should be normalized into probabilistic form first).

In PLOT, each complex concept in the upper level can be represented via the leaf concepts by the logical relations. Take an example of PLOT in figure 2, \(C^6 = C^1\) OR \(C^2\), \(C^7 = C^3\) AND \(C^4\), and \(C = (C^3\) OR \(C^2)\) OR \(C^3\) AND \(C^4)\) OR \(NOT C^5\). Such classical Boolean logic gives two exclusive results: one sample is either positive or negative for the target concept. To formulate a corresponding prior probability model for learning and inference, the Boolean logic is converted into fuzzy logic which replaces AND, OR, NOT by some continuously probability conversions. There are many different fuzzy logical operations which can do such conversion, and here we enumerate two kinds of them as follows.

Min/Max/complement:

In this case, AND is replaced by “min”, OR by “max”, and NOT by \(1 - P(y = 1 \mid f(x))\) where \(P\) is the model attached with the children node of the logic NOT. Take \(C\) in figure 2 as an example, the prior model \(P_{prior}(y \mid f(x))\) for \(C\) is

\[
P(y = 1 \mid f(x)) = \max\{\max\{P^1(y = 1 \mid f^1(x)), P^2(y = 1 \mid f^2(x))\}, \min\{P^3(y = 1 \mid f^3(x)), P^4(y = 1 \mid f^4(x))\}\}, 1 - P^5(y = 1 \mid f^5(x))\}.
\]

Probabilistic product/sum:

In this case, \(C^6 = C^1\) AND \(C^2\) is replaced by

\[
P^a(y = 1 \mid f(x)) = T_{prod}(P^1(y = 1 \mid f(x)), P^2(y = 1 \mid f(x))) = P^1(y = 1 \mid f(x))P^2(y = 1 \mid f(x)),
\]

\(C^b = C^1\) OR \(C^2\) is replaced by

\[
P^b(y = 1 \mid f(x)) = \mathcal{T}_{sum}(P^1(y = 1 \mid f(x)), P^2(y = 1 \mid f(x))) = P^1(y = 1 \mid f(x)) + P^2(y = 1 \mid f(x)) - P^1(y = 1 \mid f(x))P^2(y = 1 \mid f(x)).
\]

Again, NOT is converted by complement operation as in Case 1. The probabilistic product and sum is often called T-norm and T-conorm.

There are many other fuzzy logical operations to do a similar conversion from the classical Boolean logics to their probability forms, such as Lukasiewicz logic, Nilpotent logic and Hamacher logic. Interested readers can find more in [30].

To learn a satisfactory model for upper level concepts by PLOT, we still need to overcome the following two problems: semantic ambiguity, and error propagation.

- Semantic ambiguity. One concept can be represented in a hierarchical logical structures in several levels. There are different structures on different sub-concepts to represent the root concept. Therefore, if a hierarchical logical structure can represent a concept, a concept is not in this hierarchical logical structure yet in other hierarchical logical structure that also can represent this concept may result the semantic ambiguity. We use an example to explain this problem as illustrated in figure 3. Figure 3 illustrates a vehicle PLOT with three children nodes combining by logic OR. Since bus, car, and truck are kinds of vehicle, the samples on the positive sides of these three objects, which is the region associated with logic “bus OR car OR truck”, must also be vehicle. However, the negative sides of these objects, i.e., the ambiguity region, cannot exclude the possibility of some samples on it being vehicle. For example, an example corresponding to ship locates on the ambiguity region but it is also a vehicle. In other words, it is impossible and unnecessary to enumerate all possible kinds of subclasses of vehicles (such as ship) in PLOT. Thus the logical results conducted from PLOT cannot clarify the ambiguity in the negative sides of children nodes and some extra information is needed to clarify it. The similar problem of semantic ambiguity also exists in other logic operations.

- Error propagation. Because the probabilistic models in the nodes of low levels cannot be perfectly constructed due to incomplete semantic information and
the limitation of models. Thus the error contained in these models may be propagated into the higher-level concepts. Therefore, some relevance feedback scheme [31] is required to correct these errors when learning the higher-level concepts.

Summarizing the above two problems, in order to model high-level concepts, only using the information from the models associated with lower level nodes in PLOT is not enough. It requires some extra content-based examples to update the prior models purely obtained by the logical relations to clarify semantic ambiguity, and correct the errors from lower level models. In other words, two criterions are proposed when modeling the complex concepts in upper levels.

- **Criterion 1**: The model $P(y|f(x))$ for the upper level concepts should preserve as much information of the prior model $P_{prior}(y|f(x))$ as possible which combines the information on primitive models of lower level nodes in PLOT.

- **Criterion 2**: With the new extra training examples, the model $P(y|f(x))$ for upper level concepts must reflect the information contained in these extra training examples.

Based on the above two criteria, we formulate the proposed probabilistic algorithms to model the high-level concepts on PLOT.

### B. Learning and Inference on PLOT

Here we formally formulate our problem mathematically. Given a set of the models $P^m(y|f^m(x))$ of low-level concepts $C^m$ (where $1 \leq m \leq M$, and $M = 5$ in figure 2), as well as some extra training examples $\{x^l, y^l|1 \leq l \leq N\}$, our goal is to learn a model $P(y|f(x))$ for the target concept $C$ based on a given PLOT.

First, according to PLOT, target concept $C$ can be expanded into $C^m$ by the logical relation uncovered by PLOT. Accordingly, we can obtain a prior model $P_{prior}(y|[f^m(x)]_{m=1}^M)$ just as the example shown in figure

\[C = (C^5 \text{ OR } C^3) \text{ OR } (C^3 \text{ AND } C^4) \text{ OR } \neg C^5\]

![Fig. 2. An example of probabilistic logical ontology tree. The target concept $C$ is represented in a hierarchical logical structure in three levels. From this PLOT, $C$ is finally described by five primitive concepts in model warehouse as $C = (C^1 \text{ OR } C^2) \text{ OR } (C^3 \text{ AND } C^4) \text{ OR } \neg C^5$. It is worth noting that for each concept $C^i$, different features $f^i$ can be used as low-level descriptors.](image)

Then the new model $P(y|f(x))$ should reflect the two criteria mentioned in the end of the last subsection. Therefore, we formulate the following optimization problem to solve it,

\[
\min_{P(y|x)} \frac{1}{N} \sum_{l=1}^{N} \mathcal{D} \quad \text{s.t.} \quad \frac{1}{N} \sum_{l=1}^{N} \mathbb{E}_{P(y|f(x_l))} [y]\mathbb{E}_{P(y|f(x_l))} [y] = \frac{1}{N} \sum_{l=1}^{N} y_l + \eta
\]

\[
\sum_{y \in \{0,1\}} P\{y|f(x_l)\} = 1
\]

\[
\sum_{d=1}^{D} \frac{\theta_d^2}{2\sigma_d^2/N} + \frac{\eta^2}{2\sigma_y^2} \leq T
\]

\[1 \leq d \leq D\]
where $D = D_{KL} \left( P(y|f(x_1)) \lVert P_{\text{prior}} \left( y|\left[f^m(x_1)\right]_{m=1}^M \right) \right)$ is the Kullback-Leibler divergence, $\theta_d$ and $\eta$ are the estimation errors, $\sigma_\theta$ and $\sigma_\eta$ are two given parameters, as well as $T$ is a constant. By minimizing the divergence between $P(y|f(x))$ and $P_{\text{prior}} \left( y|\left[f^m(x_1)\right]_{m=1}^M \right)$, the information in the prior model can be preserved as much as possible according to criterion 2. $\mathbb{E}_{P(y|f(x_1))} \left[ \cdot \right]$ is the expectation with respect to the distribution $P(y|f(x_1))$ and is the $d$th element of low-level feature vector $f(x_1)$. The first two constraints in the above formulation requires the first two-order statistics of new model $P(y|f(x))$ must comply training set up, to estimate errors $\theta_d$ and $\eta$. Furthermore, the third constraint normalizes the model so that it satisfies the probabilistic property. Finally, the fourth constraint assumes the joint probability of estimation errors should be reasonably upper bounded by $T$ [17].

**Inference on PLOT:**

First, given equation (2), we see how to infer the model $P(y|f(x))$ for the target concept. From (2), the Lagrangian function is:

$$
L(P(y|f(x_1)), \theta, \eta, b, w, \gamma, \xi) =
\begin{align*}
&\mathcal{D} + \sum_{d=1}^{D} w_d \left\{ \frac{1}{N} \sum_{l=1}^{N} y_l f_d(x_l) + \theta_d - E_1 \right\} \\
&+ b \left\{ \frac{1}{N} \sum_{l=1}^{N} y_l - \eta - E_2 \right\} \\
&+ \gamma \left\{ \sum_{d=1}^{D} \frac{\theta_d^2}{2\sigma_\theta^2} + \frac{\eta^2}{2\sigma_\eta^2} - C \right\} \\
&+ \sum x_i \left( 1 - \sum y \in \{0, 1\} P(y|f(x)) \right).
\end{align*}
$$

(3)

where $E_1 = \frac{1}{N} \sum_{l=1}^{N} \mathbb{E}_{P(y|f(x_1))} \left[ y_l f_d(x_l) \right]$ and $E_2 = \frac{1}{N} \sum_{l=1}^{N} \mathbb{E}_{P(y|f(x_1))} \left[ y \right]$. By inferring the Eq. (3), we can obtain the solution of target model $P(y|f(x_1))$, as follow,

$$
P(y|f(x_1)) = \frac{1}{Z(x_1)} P_{\text{prior}} \left( y|\left[f^m(x_1)\right]_{m=1}^M \right) \cdot \mathcal{X},
$$

(4)

where

$$
Z(x_1) = \sum_{y \in \{0, 1\}} P_{\text{prior}} \left( y|\left[f^m(x_1)\right]_{m=1}^M \right) \cdot \mathcal{X}
$$

is the partition function for normalization, and $\mathcal{X} = \exp \left\{ y (w^T x_1 + b) \right\}$. The details of solving and derivative in inference can be found in Part A of Section Appendix.

**Learning on PLOT:**

Now we show how to learn $P(y|x)$, i.e., computing its model parameters $w$ and $b$. From (11) in Section Appendix we obtain

$$
\eta = \frac{-b}{\gamma N} \sigma_\eta^2,
\theta_d = \frac{-w_d}{\gamma N} \sigma_\theta^2.
$$

(5)

**Algorithm 1 PLOT**

**Input:** a set of the models $P^m(y|f^m(x))$, training examples $\{x^i, y^i|1 \leq l \leq N\}$, parameter $\theta_d$, $\eta$, $\sigma_\theta$ and $\sigma_\eta$ and $T$.

**Output:** model $P(y|f(x))$

1. Solve $P(y|f(x_1))$ with Eq. (4).
2. Solve $b^*$ and $w^*$ with Eq. (6) by conjugate gradient method.
3. $P(y = 1|f(x)) = \frac{1}{1 + e^{x_1b^*}}$.

Substitute (4) (in Section Appendix) and (5) into Lagrangian function (3), we can formulate the dual optimization problem as

$$
b^*, w^* = \arg \max_{b, w} L(P(y|f(x_1)), \theta, \eta, b, w, \gamma, \xi)
= \arg \max_{b, w} \frac{1}{N} \sum_{l=1}^{N} (y_l (w^T f(x_l) + b)) + \log P_{\text{prior}} \left( y|\left[f^m(x_1)\right]_{m=1}^M \right)
- \log Z(x_1) - \frac{\lambda_w}{2N} \|w\|^2 - \frac{\lambda_b}{2N} b^2
$$

where $\lambda_w = \frac{\sigma_\eta^2}{\gamma}$, $\lambda_b = \frac{\sigma_\theta^2}{\gamma}$ are the balance parameters. The optimization of Eq. (6) is described in Part B of Section Appendix. Algorithm 1 summarizes the algorithm steps.

**C. Efficient Online Modeling for Large Scale Problem**

As more and more visual data booms on the Internet, from image and video sharing web sites to various kinds of social communities, efficient modeling and recognition algorithms are required to handle these rising data. Among them, online modeling technique is very useful to handle the large scale data set where the samples are processed one by one. Once new data arrives, it need not re-train a brand new model with all the collected data, instead only new samples are required to update the existing model. Assume we currently have a model $P(y|f(x)) = \frac{1}{Z(x)} P_{\text{prior}} \left( y|\left[f^m(x)\right]_{m=1}^M \right) \cdot \mathcal{X}$ as equation (4) in hand, our goal is to obtain a new one $\tilde{P}(y|f(x))$ by some new samples ${\tilde{x}_i, \tilde{y}_i|1 \leq l \leq N}$. Following the similar idea in formulation (2), the new model should preserve as much information as possible in $P(y|f(x))$ as well as reflect the new information in ${\tilde{x}_i, \tilde{y}_i|1 \leq l \leq N}$. By substituting $\frac{1}{N} \sum_{l=1}^{N} D_{KL} \left( \tilde{P}(y|f(x)) \| P(y|f(x)) \right)$ into the objective function in (2) and the new examples in ${\tilde{x}_i, \tilde{y}_i|1 \leq l \leq N}$ for those in ${x_i, y_i|1 \leq l \leq N}$, we have the new model as

$$
\tilde{P}(y|f(x)) = \frac{1}{Z(x)} P_{\text{prior}} \left( y|\left[f^m(x)\right]_{m=1}^M \right)
\cdot \exp \left\{ y \left( (w + \tilde{w})^T x + (b + \tilde{b}) \right) \right\}
$$

(7)

and

$$
\tilde{Z}(x) = \sum_{y \in \{0, 1\}} P_{\text{prior}} \left( y|\left[f^m(x)\right]_{m=1}^M \right)
\cdot \exp \left\{ y \left( (w + \tilde{w})^T x + (b + \tilde{b}) \right) \right\}
$$

(8)
Where $\hat{w}, \hat{b}$ can be computed from

$$
\hat{b}^*, \hat{w}^* = \arg \max_{b, \hat{w}} \frac{1}{N} \sum_{l=1}^{N} \{ y_l (w + \hat{w})^T f(x_l) + (b + \hat{b}) \}
+ \log P_{\text{prior}} \left( y \left| f^m(x_l) \right|^M_{m=1} \right) - \log \hat{Z}(x_l) 
- \frac{\lambda_w}{2N} ||\hat{w}||^2 - \frac{\lambda_b}{2N} \hat{b}^2
$$

Since only new samples are involved in the above optimization problem, the model can be updated much more efficiently.

### D. Multimodal Feature Descriptors

Note in the learning algorithm in subsection 3.2, different feature descriptors (i.e., $f^m$ and $f$) can be used to represent the concepts associated with the nodes in PLOT. It increases the flexibility of the feature representation so that the most suitable features can be used for each concept. Moreover, with such a heterogeneous feature structure, both content-based features (i.e., visual/audio features extracted from multimedia content) and context-based features (i.e., the surrounding text, GPS location data, user tags etc. from the multimedia context) can be adopted in PLOT. Some concepts can be better modeled by the content feature such as objects (e.g., car, rocket, and horse), scenery (e.g., beach, mountain) and events (e.g., human action). On the other hand, some concepts can be better modeled by the contextual features, such as landmark place of interest (e.g., Great Wall, Eiffel tower and White House). By the above proposed learning algorithm in PLOT, different concepts can select their most suitable feature descriptors for modeling integrated in a unifying framework. However, the major contribution of the proposed work is not characterizing different features, it is still worth developing more in depth in the future.

### IV. Experiments

In this section we present experiments by comparing the proposed LEGO-MM approach with the other state-of-the-art algorithms. We demonstrate how the proposed algorithm effectively model structured concepts using model warehouse, as well as enhance target concepts classification results.

#### A. Dataset

In order to demonstrate the robustness and effectiveness of the proposed LEGO-MM approach in hierarchical visual recognition using model warehouse, we conduct experiments on ImageNet dataset [32] in the “vehicle” domain. ImageNet is a realistic image database organized by WordNet [33][18] hierarchy. Each node in the hierarchy is representing a concept, and associated with a set of images. “Vehicle” is a relatively complex root concept in ontology, including a large number of different sub-genre categories. This “vehicle” specified ImageNet subset has first been used and released in [34]. Here we adopt the same dataset to show the competitive results of the proposed LEGO-MM algorithm.

The dataset is from Concept “vehicle” in ImageNet database, which contains 26,210 images, including 13,889 positive “vehicle” samples, and 12,321 negative samples. In concept “vehicle”, there are 20 concepts associated with root “vehicle”, including a four-level ontological structure with 13 leaf nodes. Each of the leaf concept contains around 1,000 positive images. There are 20 concepts associated with root “vehicle”, including a four-level ontological structure with 13 leaf nodes. Each of the leaf concept contains around 1,000 positive images. The negative samples include concepts such as “formation”, “structure” and “sports”, which contain tremendous low level visual ambiguities. Some sample images are shown in figure 4. The “parent-child” relationship indicate a “is-a” (also seen as a OR relation in PLOT) relationship in the WordNet taxonomy. For example, “plane” in the ontology shown in figure 5 contains “OR” relationship to its children, and “Not” relationships to other nodes in the same level.

#### B. Feature Extraction and Selection

We extract the Hierarchical Gaussianization (HG) feature [35] to represent each image for our experiment. Basically, HG features jointly model appearance and spatial structures of each image by fitting into a Bayesian hierarchical framework using mixture of Gaussians. Specifically, in our experiments, we use normal HG features by first extracting 128-dimension SIFT descriptor within a 20x20 patches over a grid with five pixels spacing. Then Principal Component Analysis (PCA) is applied to each SIFT vector to reduce its dimensionality to 80. Moreover, each image is characterized by 512 Gaussian mixture components, each of which is then vectorized by a 80 dimensional vector.

In total, $80 \cdot 512 = 40,960$ dimension feature vector is extracted using PCA.

Fig. 4. Some example images in the dataset.
C. Real World Problem Simulations

In our experiments, we simulate the real-world scenario as we described in the introduction. We split all images into three disjoint sets randomly with 40%, 15%, and 45%. Then basic logistic regression models are trained using 40% of samples for the leaf nodes on the given hierarchical structure. After training, we only keep the weight vectors and use them as our “existing models” in the model warehouse. In such a way, we obtain a pool of rich semantic models for images. It is an analog to LSCOM and Mediamill 101, where all the samples used to generate the models in the pool are not accessible anymore. Actually, in real life, a large number of labeled training samples are not easy to acquire, especially when the domain of concepts expand rapidly. In order to take this fact into account, we randomly sample 15% of the data used as training images for LEGO-MM as well as other compared methods which cannot acquire information from pre-trained models. The last 45% of data is used as testing samples to evaluate recognition performance gre different approaches. All of the experiment results are reported as averaged over ten random runs.

D. Logic Operations and Error Propagation Analysis

In section III-A, we mentioned two different ways to represent Boolean logic probabilistically. One way is using Min/Max function, the other way is using T-norm/T-conorm. Moreover, we also illustrated how important of the small number of training samples to prevent error propagation from lower level to higher level. Here we compare the classification results on proposed LEGO-MM approach and a purely logical version without using any updating samples called P-LEGO-MM. And the results is shown in table I.

Two main observations can be made from table I. First, both probabilistic logic operations provide comparable results on all different concepts regardless of how far these concepts are away from the “existing models”. In general, “T-conorm” performs slightly better than “Max” in both LEGO-MM and P-LEGO-MM methods on all the eight different target concepts. Second, P-LEGO-MM’s accuracy decreases dramatically as the “target concept” comes higher in the ontology hierarchy. On the other hand, our proposed LEGO-MM algorithm still has a reliable performance. The main reason is that the general procedure using logical operations on hierarchical structure depends on lower level concepts that provide information to learn “target concepts” at higher levels. Once the prior is obtained, model adaptation process will start. In the meantime the prior probabilities for its parent’s nodes are also computed in the same manner. However, without model-updating scheme, higher level models cannot be reliably refined and it could result in errors accumulated through the entire hierarchical structure. A similar observation also can be obtained from figure 6, the two ROC curves in the first row illustrates the classification accuracy at the 3rd level, and the two ROC at the bottom is the 2nd level concepts.

E. Performance Comparison

In previous section we have shown that only combining primitive concepts is insufficient due to error propagations. To evidently demonstrate the advantage of the proposed LEGO-MM method for ontological categorizations, we compare experimental results with other state-of-the-art approaches. The comparison experiments strictly follow the protocol in section IV-C, and the results are shown in table II. In [34] and [35], the authors proposed the best classifier using [35]’s feature was obtained by first applying Within Class Covariance Normalization (WCCN), and then using Nearest Neighbor or Nearest Central (WCCN+NN, and WCCN+NC). In our setup, both methods yield comparable results as flat multi-class SVM [2]. However, there is no straight way to integrate these approaches in the given ontology hierarchy. Therefore, we also conduct our experiment using a tree loss based hierarchical SVM proposed in [36] [37]. Besides, we also add the comparison of deep convolutional neural networks, which is the state-of-the-art classification method in recent years. Among all, our proposed LEGO-MM algorithm provides a significant gain over the typical methods, especially at the 3rd level of classification. By using the CNN features as the image descriptors (called Ours + CNN features), the performance of our proposed LEGO-MM is further improved over DCNN. This shows that the “existing models” in the model warehouse provide tremendous contributions to distinguishing more abstractive categories at higher level.

Fig. 5. Vehicle hierarchy from ImageNet. The red color notes indicate the leaf-node concepts, which can be obtained from the model warehouse.
F. Robustness and Online Efficiency

To test the robustness of our proposed algorithm, we first look into the case of how a limited number of primitive concepts affect performances. Second, we study the sensitivity of learning rate of our approach. The last but not the least, online and batch comparison will be reported.

1) Incomplete Model Warehouse: In practice, when we model a structured concept hierarchy, even if we could access a rich semantic pool, it might still be insufficient. Because not all the possible sub-genres of “target concepts” can be covered in the warehouse. To illustrate such more challenging scenario with incomplete model warehouse, we take away some leaf-node concepts from the primitive pool before we have all leaf node “existing models’” accessible, now we are only allowed to access some of them. Figure 7 demonstrates how LEGO-MM algorithm handles the case of incomplete model warehouse. Three experiments are performed: the leftmost histogram illustrates the classification accuracy for concept “plane” at 3rd level, and “aerial” at 2nd without the prior model “warplane.” Similarly, the middle and the right histograms are both missing a leaf-node concept, which is “bicycle”, and “sailboat.” We report the recognition performances on the parent and grandparent nodes. Since less prior information is able to be obtained by LEGO-MM, it will introduce a huge semantic ambiguity and the middle and the right histograms are both missing a leaf-node concept. Since less prior information is able to be obtained by LEGO-MM, it will introduce a huge semantic ambiguity. We report the recognition performances on the parent and grandparent nodes. Since less prior information is able to be obtained by LEGO-MM, it will introduce a huge semantic ambiguity.

2) Learning Rate Trade off: In equation (6) we introduced the learning rate $\lambda$ as a balance parameter between criteria 1 and 2 mentioned in section III-A. Figure 8 illustrates how such $\lambda$ affects recognition performance. Seven curves in Figure 8 indicate how the accuracy on each intermediate nodes changes as learning rate increases from 1 to 800. The recognition performances tend to stabilize after $\lambda$ goes greater than 150. Thus, we can set the learning rate to 200 and obtain reliable results in the experiments.

3) Online Step Size: Another parameter affecting the online modeling of the proposed LEGO-MM algorithm is the step size, which trades off between the modeling performance and the number of updating samples involved in each step. Figure 9 demonstrates the relationship between the...
step size (in horizontal axis) and classification performance (in vertical axis) with a fixed learning rate. The online updating with a small step size (e.g., only one) performs the worst, because at each step, much fewer samples can result in the risk of introducing huge variance to the model. On the contrary, when the step size reaches to about 200, the performance becomes stable. After that, even increasing the step size does not affect the overall performance too much. In all means, the proposed LEGO-MM achieves much robust performance with an online updating strategy.

G. Extension: Multimodal Features and Single Features

To illustrate the advantage of multimodal feature descriptors, we further conduct the experiments of the proposed method with single-modal features and multi-modal features. The dataset is Concept "skating" in ImageNet database, which contains 9,806 images. As shown in figure 10, there are 7 concepts associated with root "skating", including a three-level ontological structure with 4 leaf nodes. Each of the leaf concept contains 500 positive images. The negative samples include concepts such as “formation”, “structure” and “vehicle”, which contain tremendous

![ROC curve for Plane](image1)

![ROC curve for Ship](image2)

![ROC curve for Aerial](image3)

![ROC curve for Marine](image4)

Fig. 6. ROC curve for concept “plane” and “ship” at the 3rd level, and “aerial” and “marine” at the 2nd level.
low level visual ambiguities. The comparison experiments strictly follow the protocol in section IV-C. The multi-modal features are SIFT features and CNN features [38], [39]. The former is one of the most classical features, while the latter is the state-of-the-art image feature descriptors. The results is shown in Table III. We can see that the proposed method with multi-modal features gains the better performance than the proposed method with single-modal features.

TABLE III

<table>
<thead>
<tr>
<th>Feature</th>
<th>2nd level</th>
<th>3rd level</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ours with SIFT features</td>
<td>82.18</td>
<td>82.36</td>
</tr>
<tr>
<td>Ours with CNN features</td>
<td>89.47</td>
<td>90.20</td>
</tr>
<tr>
<td>Ours with SIFT + CNN features</td>
<td>90.35</td>
<td>91.08</td>
</tr>
</tbody>
</table>

V. CONCLUSION

In this paper we developed a novel framework, LEGO-MM, to seamlessly integrate both the new target training examples and the existing primitive concept models. LEGO-MM treats the primitive concept models as “lego” to potentially construct an unlimited vocabulary of new concepts. We proposed a much flexible learning algorithm to efficiently combine the obtained probabilistic model with new information to clarify semantic ambiguity as well as correct the errors propagated from the nodes at lower level. The extensive experiments over a real-world data set demonstrated: 1) using logical operations combining individual concepts in the existing model warehouse provides us with rich semantic resources to improve performance on more abstractive concepts at higher level of ontology hierarchy; 2) evolving higher level models by using a small number of examples could clarify the semantic ambiguities. Particularly, our proposed “T-conorm” LEGO-MM approach has significant advantages over other state-of-the-art algorithms; 3) LEGO-MM is also robust with incomplete concepts in the existing model warehouse, varying learning rates and online step sizes.

ACKNOWLEDGMENTS

This work was partially supported by the 973 Program of China (Project No. 2014CB347600), the National Natural Science Foundation of China (Grant No. 61522203 and 61402228), and the National Ten Thousand Talent Program of China (Young Top-Notch Talent).

VI. APPENDIX

A. Part A

By deriving the Eq. (3) with respect to \( P(y|f(x_i)) \) and setting the results to be zero, we have:

\[
\frac{\partial \mathcal{L}}{\partial P(y|f(x_i))} = \frac{1}{N} \left( \log P(y|f(x_i)) + 1 - \log P_{\text{prior}} \left( y \left| \left[ f^m(x_i) \right]_{m=1}^M \right) \right) - y(w^T x_i + b) \right) - \xi(x_i) = 0,
\]

and

\[
\frac{\partial \mathcal{L}}{\partial \eta} = b + N\gamma \frac{\eta}{\sigma^2} = 0,
\]

\[
\frac{\partial \mathcal{L}}{\partial \theta_d} = w_d + N\gamma \frac{\theta_d}{\sigma^2} = 0.
\]

From (10) we have:

\[
P(y|f(x_i)) \propto P_{\text{prior}} \left( y \left| \left[ f^m(x_i) \right]_{m=1}^M \right) \right) \exp \left\{ y(w^T x_i + b) \right\}.
\]

Now, considering the normalization constraints in 2, which ought to be

\[
P(y|f(x_i)) = \frac{1}{Z(x_i)} P_{\text{prior}} \left( y \left| \left[ f^m(x_i) \right]_{m=1}^M \right) \right) \exp \left\{ y(w^T x_i + b) \right\},
\]

where

\[
Z(x_i) = \sum_{y \in \{0, 1\}} P_{\text{prior}} \left( y \left| \left[ f^m(x_i) \right]_{m=1}^M \right) \right) \exp \left\{ y(w^T x_i + b) \right\}
\]

is the partition function for normalization. Thus we have obtained the target model as shown in equation (13), where the corresponding concept \( C \) can be inferred.

B. Part B

This maximization problem in Eq. (6) is unconstrained convex problem with respect to \( b \) and \( w \), so a global
maximum exists. Take the derivatives with respect to \( b \) and \( w \), we have

\[
\frac{\partial L}{\partial b} = \frac{1}{N} \sum_{i=1}^{N} y_i - \frac{1}{N} \sum_{i=1}^{N} \left( \mathbb{E}_{y \mid x} y_i \right) - \frac{\lambda_0 b}{N}
\]

\[
\frac{\partial L}{\partial w_l} = \frac{1}{N} \sum_{i=1}^{N} y_i f_l(x_i) - \frac{1}{N} \sum_{i=1}^{N} \left( \mathbb{E}_{y \mid x} y_i \right) f_l(x_i) - \lambda_w w_l / N
\]

Then (6) can be maximized by a conjugate gradient method based on (6) and its derivatives in Eq. (14).

REFERENCES

Jinhui Tang is a Professor in School of Computer Science and Engineering, Nanjing University of Science and Technology, China. He received his B.E. and Ph.D. degrees in July 2003 and July 2008 respectively, both from the University of Science and Technology of China. From 2008 to 2010, he worked as a research fellow in School of Computing, National University of Singapore. His current research interests include large-scale multimedia search. He has authored over 100 journal and conference papers in these areas. Prof. Tang is a recipient of ACM China Rising Star Award and a co-recipient of the Best Student Paper Award in MMM 2016, and Best Paper Award in ACM MM 2007, PCM 2011 and ICIMCS 2011.

Shiyu Chang received the B.S. and M.S. degrees from the University of Illinois at Urbana-Champaign, in 2011 and 2014, respectively, where he is currently pursuing the Ph.D. degree under the supervision of Prof. T. S. Huang. He has a wide range of research interests in data exploratory and analytics. His current directions lie on building high performance and reliable systems with the help of large-scale multimodality information to solve complex computational tasks in real world. He received the Best Student Paper Award in the IEEE ICDM 2014.

Guo-Jun Qi is an assistant professor in the Department of Computer Science at University of Central Florida. He received the Ph.D. degree in Electrical and Computer Engineering from the University of Illinois at Urbana-Champaign. His research interests include pattern recognition, machine learning, computer vision and multimedia. He was the co-recipient of the best student paper award in IEEE Conference on Data Mining (2014), and the recipient of the best paper award (2007) and the best paper runner-up (2015) in the ACM International Conference on Multimedia. He has served or will serve as program co-chair of MMM 2016, an area chair of ACM Multimedia (2015, 2016), a senior program committee member of ACM CIKM 2015 and ACM SIGKDD 2016, and program committee members or reviewers for the conferences and journals in the fields of computer vision, pattern recognition, machine learning, and data mining. Dr. Qi has published over 60 academic papers in these areas. He also (co-)edited the two special issues on IEEE transactions on multimedia and IEEE transactions on big data.

Qi Tian received the BE degree in electronic engineering from Tsinghua University, China, in 1992, the MS degree in electrical and computer engineering from Drexel University in 1996, and the PhD degree in electrical and computer engineering from the University of Illinois, Urbana-Champaign, in 2002. He is currently a professor in the Department of Computer Science, University of Texas at San Antonio. He took a one-year faculty leave at Microsoft Research Asia (MSRA) during 2008-2009. His research interests include multimedia information retrieval and computer vision. He has published more than 280 refereed journal and conference papers. He received the Best Paper Awards in PCM 2013, MMM 2013, and ICIMCS 2012, the Top 10% Paper Award in MMSP 2011, and the Student Contest Paper Award in ICASSP 2006. He is on the editorial board of the IEEE Transactions on Multimedia, IEEE Transactions on Circuit and Systems for Video Technology, Multimedia Systems Journal, Journal of Multimedia, and Journal of Machine Visions and Applications.

Yong Rui received the B.S. degree from Southeast University, the M.S. degree from Tsinghua University, and the Ph.D. degree from the University of Illinois at Urbana-Champaign. He is currently the Deputy Managing Director with Microsoft Research Asia (MSRA), leading research groups in multimedia search and mining, and big data analysis, and engineering groups in multimedia processing, data mining, and software/hardware systems. He has authored 16 books and book chapters, and more than 100 refereed journal and conference papers. His publications are among the most cited 15000+ citations and his h-index of 52. He holds 60 issued U.S. and international patents. He is a fellow of IAPR and SPIE, a Distinguished Scientist of ACM, and a Distinguished Lecturer of both ACM and IEEE. He is an Executive Member of ACM SIGMM, the Founding Chair of its China Chapter. He is recognized as a leading expert in his research areas. He is the Editor-in-Chief of the IEEE Multimedia Magazine, an Associate Editor of the ACM Transactions on Multimedia Computing, Communication and Applications, a Founding Editor of the International Journal of Multimedia Information Retrieval, and a Founding Associate Editor of the IEEE ACCESS. He was an Associate Editor of the IEEE TRANSACTIONS ON MULTIMEDIA (2004-2008), the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGIES (2006-2010), the ACM/Springer Multimedia Systems Journal (2004-2006), and the International Journal of Multimedia Tools and Applications (2004-2006). He also serves on the Advisory Board of the IEEE TRANSACTIONS ON AUTOMATION SCIENCE AND ENGINEERING.

Thomas S. Huang received his Sc.D. from MIT in 1963. He is a fulltime faculty with Beckman Institute at University of Illinois at Urbana-Champaign. He was William L. Everitt Distinguished Professor in the Department of Electrical and Computer Engineering and the Coordinated Science Lab (CSL), and was a full-time faculty member in the Beckman Institute Image Formation and Processing and Artificial Intelligence groups. His professional interests are computer vision, image compression and enhancement, pattern recognition, and multimodal signal processing.