Semi-Supervised Spatial Knowledge Transfer with Deep Generative Models

by

Kousuke Ariga

Submitted to the
Paul G. Allen School of Computer Science & Engineering
in partial fulfillment of the requirements for the degree of

Bachelor of Science in Computer Science with Honors
at the
University of Washington
June 2017

Author ..........................................................
Paul G. Allen School of Computer Science & Engineering
June 1, 2017

Certified by ..................................................
Dr. Andrzej Pronobis, Thesis Advisor
Abstract

We have witnessed rapid advancements in the fields of robotics and artificial intelligence over the last several years, and we expect an increasing presence of mobile robots in our daily lives. To create a world where humans and robots are able to co-work safely and effectively, it is important to have robots learn our semantic understanding of environments, or obtain spatial knowledge.

The recent success of deep learning motivates us to apply its methods to the robot’s spatial knowledge learning, but the challenge is that there are not enough labeled data to successfully generalize various environments. Unlike the collection of images or texts on the web, collecting labeled data generated by mobile robots is generally more expensive.

This thesis proposes a method to make robots generalize spatial knowledge in a semi-supervised manner. Specifically, we train a deep generative model with a few pieces of labeled data and then leverage a larger set of unlabeled data to adapt the previously learned knowledge to a different unknown environment, i.e., transfer the knowledge learned in one environment to another.

Our experimental evaluation demonstrates that our method achieves effective spatial knowledge transfer, can successfully acquire general semantic understanding of a real environment incrementally without supervision, and classifies places where a robot is located into semantic categories. The following evaluations show the effectiveness of the learning algorithm and applicability to real-world problems where a large amount of labeled data does not always exist.
# Contents

1 Introduction ......................................................... 4

2 Related Work & Background ......................................... 6
   2.1 Sum-Product Networks ........................................... 7
   2.2 Deep Generative Spatial Models ................................. 8

3 Semi-Supervised Spatial Knowledge Transfer .................. 10
   3.1 Problem Formalization .......................................... 10
   3.2 Methodology .................................................... 11

4 Experimental Evaluation ............................................ 14
   4.1 Datasets .......................................................... 14
      4.1.1 Local Environment Representation ........................ 15
   4.2 Experiments ..................................................... 17
   4.3 Results .......................................................... 18

5 Discussion & Future Work ........................................... 19

6 Conclusion .......................................................... 21
Chapter 1

Introduction

We have been longing for a robot that can robustly perform human-like tasks in unstructured environments; however, the dream has not yet come true. The fundamental problem is the difference between how robots and humans perceive the environment. Robots see the world through sensors such as cameras and rangefinders that receive the reflections of light from the surrounding objects and determine their direction, distance, and color collectively. This information allows robots to build an occupancy grid map of the environment and avoid obstacles while navigating to goals, but such a primitive representation of the world is not enough for robots to perform human-like general tasks. Humans usually rely on a more abstract representation of the environment. For example, when we want a cup of coffee, we infer that it would be available in a kitchen and go there directly. Note that we used semantic information on the environment, *kitchen*, here. A naively built robot would need to search through a building for the sensor input that forms a coffee cup because it has no semantic clues. Robots can better perform real-world tasks in human environments by leveraging spatial knowledge at all levels of abstraction [1].

Semantic mapping is an approach to model spatial knowledge at multiple levels of abstraction from data generated by the robot’s sensors, through geometry and appearance up to high-level semantic descriptions. Amongst various models that have been created to model spatial knowledge, the recently proposed deep generative spatial models (DGSM) based on a relatively new graphical model, sum-product networks
(SPNs), has shown considerable potential. Yet DGSM has the same shortcoming as most of today’s deep machine learning models. Deep learning approaches generally require a large set of labeled data to train the model for successful generalization [2]. For challenging tasks like learning a unified model of an environment that spans multiple levels of abstraction, the amount of data required to train the model is enormous. Furthermore, in many robotics domains, the collection of training data is expensive because of the slow robotic hardware and necessity of human supervision. Therefore, it is often the case that the time required for real-world experiments is the bottleneck to successfully apply rigorous deep learning techniques to the robotics domain.

To mitigate the problem of data scarcity, we investigated the use of semi-supervised learning on DGSM to leverage unlabeled data, which are relatively more affordable. Unlike supervised learning, which only uses labeled data, semi-supervised learning makes use of unlabeled data to train machine learning models. In this work, we first created a fully annotated large dataset containing rangefinder-based scans representing the robot’s local environment. Then, we investigated the effect of semi-supervised learning on DGSM through place categorization tasks with multiple configurations. We show that the additional unlabeled data helps a robot to generalize the spatial knowledge and better classify unknown places into semantic categories. Overall, the contributions of this work are as follows:

- A novel method for spatial knowledge transfer that leverages semi-supervised learning on a deep generative model [Chapter 3, Section 4.3].

- A large-scale dataset collected at three different real buildings that contains the rangefinder-based local representation of the environment and estimated robot poses [Section 4.1].
Chapter 2

Related Work & Background

Semantic mapping has been widely studied with different tasks and settings [6], and our experiments can be categorized as the place categorization task in a large-scale indoor environment. In a similar setting, discriminative models including support vector machine (SVM) have been studied and shown competent results [7]. However, discriminative models cannot represent the likelihood of data, making it impossible to leverage unlabeled data by our method. A recently proposed model for representing spatial knowledge, DGSM, is a generative model that is able to represent a joint probability distribution, giving wider capability including missing value completion and novelty detection, as well as our likelihood-based semi-supervised learning method.

Leveraging unlabeled data on top of a limited amount of labeled data is a well-known method that improves learning performance. This approach is especially useful in domains where labeled data are difficult to collect but unlabeled data are available relatively easily. Robotics is a great example because collecting labeled sensory data on real robots is expensive because of the necessity of human supervision and the slowness of the hardware. This problem is generally studied as semi-supervised learning. There are mainly two approaches for semi-supervised learning: using labeled data to learn the model and label the unlabeled data [8], which our method involves, and using regularizing side information such as the structure of the model [15]. There is also a new approach that trains generative models by using the unknown label as a missing value [5]. Semi-supervised learning has been applied to some
robotics domains, such as object tracking [14] and controls [3], but to our knowledge, applications to spatial knowledge learning have not been investigated yet.

In the rest of this chapter, we will briefly describe DGSM and SPNs, which are relatively new, and construct the foundation of our work.

2.1 Sum-Product Networks

SPNs are a graphical model with a solid theoretical foundation and several appealing properties such as fast and exact probabilistic inference over deep layers [10] [4] [9]. One of the challenges of traditional graphical models is the complexity of their partition function. SPNs handle it by constraining their structure to directed acyclic graphs with variables as leaves, sums and products as internal nodes, and summing coefficients as weighted edges [10]. The catch is that SPNs cannot encode all probability distributions; however, recent experiments have demonstrated that the class of distributions polynomial-sized SPN span is large enough for many problems.

In SPNs, the sum nodes represent mixtures over sub-models recursively at each level with their weights as the mixture prior. This enables SPNs to represent hierarchical concepts. We can regard such mixtures as latent variables and infer their value by an upward pass through the network. SPNs take the indicator variables at their leaves and output its probability as the value of the root node.

MPE inference can be performed [10] by first replacing all sum nodes with max nodes and setting the indicators of the variables for which the MPE state is inferred as 1. Then, by repeating the upward pass and downward pass, which selects the max node.

Figure 2-1: A simple naive Bayes mixture model with three components over two binary variables represented as an SPN. The bottom layer represents indicators for each of the two variables. $Y_1$ represents a latent variable marginalized out by the top sum node. Adapted from Pronobis & Rao, 2017 [12] with permission.
for which the sum node in place had the highest weight, the MPE state of the indicator variables are selected. The indicators selected by this process indicate the MPE state of the variables. SPNs can be learned both generatively [10] and discriminatively [4] using expectation maximization or gradient descent.

2.2 Deep Generative Spatial Models

DGSM is a recently proposed model of spatial concepts based on SPNs, and it learns directly from raw sensory data[12][13]. It unifies the layers of representation of environments, enabling upward and downward inferences about spatial concepts defined at different levels of abstraction. The model learns the probability distribution

\[ P(C, D^p_1, \ldots, D^p_{N_p}, D^{v_1}_1, \ldots, D^{v_8}_{N_v}, X_1, \ldots, X_{N_x}) \]

where \( C \) represents the semantic category of a place, \( D^p_1, \ldots, D^p_{N_p} \) constitute an internal descriptor of the place, \( D^{v_1}_1, \ldots, D^{v_8}_{N_v} \) are descriptors of eight views, and \( X_1, \ldots, X_{N_x} \) are input variables representing the occupancy in each cell of the local occupancy grid.

In an occupancy map built by robots, each point is represented by three indicators for empty, occupied, or unknown space. These indicator variables are used as the leaves of the SPN network. The structure of the model is generated randomly using the algorithm described in [12].

As explained in detail in [12], the resulting model is a single SPN with three levels of conceptual hierarchy. In the bottom level, we split the polar grid into eight views by

Figure 2-2: The SPN structure implementing DGSM. Adapted from [12] with permission.
45 degrees. For each view, we randomly generate a sub-SPN over the subset of $X_i$, representing the occupancy within the view, as well as latent variables $D$ serving as an internal view descriptor. In the second level, we use the distributions defining the components from each view as inputs and generate random SPNs representing each of the semantic place classes. Each of such SPNs is itself a mixture model with the latent variable being part of the place descriptor. Finally, in the third level, the sub-SPNs for place classes are combined by a sum node forming the root of the whole network. The latent variable associated with the root node is $C$ and is set to the appropriate class label during learning. Overall, such decomposition allows us to use networks of different complexity for representing lower-level features of each view and for modeling the top composition of views into place classes.
Chapter 3

Semi-Supervised Spatial Knowledge Transfer

3.1 Problem Formalization

We first define semi-supervised spatial knowledge transfer. Our method is general and can be applied to a wide range of machine learning tasks, but here we focus on the mobile robotics application. Consider situations where only a limited amount of labeled training data is available; i.e., we can barely supervise our agent, but we want it to keep learning spatial knowledge in a much larger set of unlabeled data after the initial training.

For example, if the task corresponds to an autonomous mobile robot learning the spatial knowledge of buildings to navigate intelligently, the labeled sensory data might correspond to the sensory data that was recorded and labeled at the factory using a mock building, while the unlabeled data might be the new data obtained after deployment at the customer’s building. We want the robot to keep learning and acquire general knowledge about the environment as it experiences more spaces.

Formally, we consider a distribution $P(Y, X_1, \ldots, X_C)$ over spatial knowledge of a common environment using DGSM, where $Y$ is the semantic category of a local place and $X_i$ are the input variables from the sensors as discussed in Section 2.2. Let $L$ and $U$ denote two sets of sensory data sampled from the distribution $P$. Learning can be
done on both sets; i.e., robots may increase the spatial knowledge in any environment, but supervision is given only in the set of $\mathcal{L}$. The objective of semi-supervised spatial knowledge transfer is to model a transferable general spatial knowledge about common environments $P(Y, X_1, \ldots, X_C; \theta)$ by learning the vector of parameters $\theta$ from data $\mathcal{U}$ on top of the data $\mathcal{L}$. The learning objective is

$$\theta^* = \arg\max_\theta \prod_{y, x_1, \ldots, x_C \in \mathcal{D}} P(y, x_1, \ldots, x_C; \theta)$$

where $\mathcal{D} \subseteq \mathcal{L} \cup \mathcal{U}$, and the inference objective is

$$y^* = \arg\max_y P(y|x_1, \ldots, x_C; \theta).$$

By incorporating both labeled and unlabeled data, we can develop an algorithm that alternates between learning spatial knowledge and inferring the class of the unlabeled data. However, the order, amount, and selection of the unlabeled data to feed to the learning algorithm have to be designed carefully to provide robots with a good curriculum so that they can generalize the spatial knowledge effectively.

### 3.2 Methodology

We now present our approach for performing semi-supervised spatial knowledge transfer with deep generative models. As discussed previously, our goal is to train DGSM that can better infer the class of the local environment by using additional unlabeled data in a different distribution.

We propose three algorithms. Suppose there are three sets of data $\mathcal{L}$, $\mathcal{U}_{1\ldots n}$, $\mathcal{T}$ where $\mathcal{L}$ is the labeled training dataset and $\mathcal{U}$ is the unlabeled training dataset as discussed in the previous section. The unlabeled dataset may be generated from an arbitrary number of different environments. $\mathcal{T}$ denotes the dataset by which the semantic understanding of the environment is tested.

The first algorithm uses all the available data in $\mathcal{U}$ for training sub-SPNs no
Algorithm 1 Semi-Supervised Spatial Knowledge Transfer

1: $S \leftarrow \text{GenerateDenseSPN}(\mathcal{L}, \mathcal{U}, \mathcal{T})$
2: $\text{InitializeWeights}()$
3:
4: repeat
5:     for all $l \in \mathcal{L}$ do
6:         $\text{UpdateWeights}(S, \text{Inference}(S, l))$
7: until convergence
8:
9: for $i = 1, \ldots, n$ do
10:     repeat
11:         for all $u \in \mathcal{U}_i$ do
12:             if $\text{Inference}(S, u) > \text{threshold}$ then
13:                 $\text{UpdateWeights}(S, \text{Inference}(S, u))$
14:         until convergence
15:
16: repeat
17:     for all $t \in \mathcal{T}$ do
18:         if $\text{Inference}(S, t) > \text{threshold}$ then
19:             $\text{UpdateWeights}(S, \text{Inference}(S, t))$
20: until convergence
21:
22: $S \leftarrow \text{PruneZeroWeights}(S)$

matter which class they represent. We expect that even when the distribution of the
data does not match the class that a sub-SPN represents, a large amount of data still
effectively trains the lower layer of the SPNs. In other words, through a large set of
unlabeled data, DGSM may be able to generalize the lower-level geometric structure
of the environment independently from the class label.

The second algorithm uses a subset of data $\mathcal{D} \subseteq \mathcal{U}$ such that the likelihood of all
d $\in \mathcal{D}$ is greater than a certain threshold. By doing so, we can only use the unlabeled
data that are likely in the category that the model is trying to learn.

The third algorithm leverages even the testing data $\mathcal{T}$. Imagine that a robot is
deployed in an unknown environment. Some of the places may look like the place
that the robot knows from previous experiences, and some may not. Suppose the sizes
of offices are generally larger than the offices that the robot already knows. In this
scenario, if the robot could find a few smaller instances of offices in the new building,
it can infer that the room is an *office* although it is a little larger than expected. Then, in the next iteration of learning, the robot can infer the slightly larger rooms as *office* and use them for training as labeled data. By repeating this procedure, robots can incrementally enlarge the set of places that they can use for self-training.

To learn DGSM, we basically use the *learnSPN* algorithm [10] but with a modification to accommodate semi-supervised learning. The full algorithm for the third one is presented in Algorithm 1. The first and second algorithms can be derived easily from the pseudo code for the third algorithm. We can run the second algorithm by simply skipping the third loop block using the testing data $T$ and the first algorithm by additionally ignoring the threshold in the second block. We used hard EM and marginal inference for the weight update and inference accordingly.
Chapter 4

Experimental Evaluation

Our experiment evaluated the model in the typical scenario of semantic place categorization. We will describe our datasets and the experimental configurations in this chapter.

4.1 Datasets

Our experiments were performed on rangefinder-based data from the COLD database [11]. The COLD database contains the robot’s odometry and laser range data captured using a mobile robot navigating through multiple floors of buildings in Saarbrucken and Freiburg. On top of this database, we have similar data obtained in Stockholm. For our experiments, we derived the robot’s estimated pose and a rangefinder-based representation of the local environment, which is described in detail in the next section. We will publicize these additional data soon. It is fully annotated and synced by time, so we expect that our dataset can be used for a wide range of robotics research.

The places are labeled with their semantic categories with high granularity, such as large office and professor office. Also, for some categories, few instances are available, such as printer area and kitchen. In our experiment, we merged multiple types of offices into office and mapped all the labels into four categories: corridor, doorway, office, and unknown for simplicity. Refer to Table 4.1 for the details.
Despite the simplification, the sizes of multiple types of offices differ by factor of two or more. Also, the local appearance of corridors and doorways varies substantially in different parts. As a result, there is significant variability within each place category. The distribution of the resulting place categories for each building is shown in Figure 4-1.

Figure 4-1: The distribution of place categories in each building.

### 4.1.1 Local Environment Representation

We use occupancy grid maps generated from laser range data to represent an observation of the local environment. Our goal is to model the spatial knowledge of local
environments; therefore, we constrain the observation to the parts that are visible from the robot. To this end, we process the global occupancy grid as described in Figure 4-2. We first filter small obstacles and fill holes on walls, and then we raytrace from the robot’s position. As a result, the local map will mostly contain objects present inside a single room when the robot is in a room.

Then, all local observations available to the robot are transformed into a robocentric polar occupancy grid representation as shown in Figure 4-3. As explained in [12], the resulting observation contains higher-resolution details closer to the robot.
and lower-resolution information further away. This focuses the attention of the model to the nearby objects. Higher resolution of information closer to the robot is important for understanding the semantics of the robot's exact location (for instance when the robot is in a doorway). However, it also relates to how spatial information is used by a mobile robot while executing actions. It is in the vicinity of the robot that higher accuracy of spatial information is required. A similar principle is exploited by many navigation components, which use different resolutions of information for local and global path planning. Additionally, such a representation corresponds to the way the robot perceives the world because of the limited resolution of its sensors. The polar grids in our experiments were built for the maximum radius of 5m, with an angle step of 6.4 degrees and resolution decreasing with distance from the robot.

### 4.2 Experiments

We conducted three experiments, varying the amount and quality of the unlabeled data given to the model for training as described in Section 3.2. We initialize the model using the data collected at Freiburg and tested the model using the data obtained at Saarbrucken in all the experiments. In the first experiment, we train the model using all the Stockholm data without labels. In the second experiment, we similarly train the model using the unlabeled Stockholm data, but we threshold them by their likelihood (we used -500 in this experiment) in every epoch of EM learning. Last, in the third experiment, on top of the second experiment, we retrain the model using the testing data without labels.

<table>
<thead>
<tr>
<th>Experiment Settings</th>
<th>Experiment Settings</th>
<th>Experiment Settings</th>
<th>Experiment Settings</th>
<th>Experiment Settings</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Phase 1</td>
<td>Phase 2</td>
<td>Phase 3</td>
<td>Testing</td>
</tr>
<tr>
<td></td>
<td>(labeled)</td>
<td>(unlabeled)</td>
<td>(unlabeled)</td>
<td></td>
</tr>
<tr>
<td>Baseline</td>
<td>Freiburg</td>
<td>None</td>
<td>None</td>
<td>Saarbrucken</td>
</tr>
<tr>
<td>Experiment 1</td>
<td>Freiburg</td>
<td>Stockholm</td>
<td>None</td>
<td>Saarbrucken</td>
</tr>
<tr>
<td>Experiment 2</td>
<td>Freiburg</td>
<td>Stockholm*</td>
<td>None</td>
<td>Saarbrucken*</td>
</tr>
<tr>
<td>Experiment 3</td>
<td>Freiburg</td>
<td>Stockholm*</td>
<td>Saarbrucken*</td>
<td>Saarbrucken</td>
</tr>
</tbody>
</table>

Table 4.2: * Only the data whose likelihood is greater than −500 were used.
4.3 Results

Figure 4-4: Normalized confusion matrix for the problem of semantic categorization of local observations.

We see that the algorithm 2 demonstrated the best categorization performance. It implies our method successfully generalized spatial knowledge using unlabeled data. Most of the confusion exists between corridor and others. This is understandable given that corridors in the dataset varies significantly across buildings. The algorithm 1 showed worse result than the baseline. Also, the algorithm 3 did not improve the success rate compared to the algorithm 2 although it is trained on the unlabeled test data on top of the algorithm 2. We will discuss the results in the next chapter.

<table>
<thead>
<tr>
<th>Methods</th>
<th>Success Rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>Baseline</td>
<td>61.23%</td>
</tr>
<tr>
<td>Algorithm 1</td>
<td>44.96%</td>
</tr>
<tr>
<td>Algorithm 2</td>
<td>67.55%</td>
</tr>
<tr>
<td>Algorithm 3</td>
<td>67.47%</td>
</tr>
</tbody>
</table>

Table 4.3: Average success classification rate.

Figure 4-5: Transition of the ratio of unlabeled data that was used for the training at phase 2 and 3.
Chapter 5

Discussion & Future Work

The experiment 2 demonstrated that the proposed method successfully transferred the spatial knowledge across multiple buildings and improved the semantic categorization performance in the semi-supervised manner. However, the experiment 1 and 3 revealed its limit, too. Specifically, it is difficult to choose the threshold.

In the experiment 1, we used all the unlabeled data at the phase 2, and the success categorization rate was degraded. This result suggests that when the samples used for the semi-supervised learning are mixed with many samples in wrong categories, i.e. when the training data is too noisy, DGSM confuses itself, contrary to our expectation that the model would learn the low-level spatial features regardless of its category. Therefore, it is important to assure the correctness of the unlabeled data that is used at the phase 2 and 3 by using a threshold that gives low false positives.

We chose the threshold $-500$ somewhat arbitrarily with a reasoning that if the likelihood of a sample generated from a learned model is as high as $-500$, then the sample is likely from the category that the model represents according to the empirical result in [12]. At the phase 2, this threshold successfully extracted corridor and door samples from the unlabeled dataset and increased the success rate for these categories, however, it excluded all the office samples, resulting in a slightly worse success rate. On the other hand, at the phase 3, only the office samples are used for semi-supervised training, and not the others. We could fine tune the threshold value, however, it would overfit the model to our specific dataset.
If we can extract the subset of unlabeled data that represents the category that we are training with low false positive rate, then our method will be much more effective. A immediate possibility is to use an unsupervised clustering algorithm to cluster the unlabeled dataset and use the data in the cluster that is most similar to the target category for the semi-supervised training. However, this method would also fail when the category does not exist in a new environment since we need to determine whether we use the obtained cluster for the training or not by applying certain threshold.

Wisely selecting the threshold is the corner stone of the semi-supervised learning, and a novel effective method would significantly improve our semi-supervised spatial knowledge transfer algorithm. We will explore this direction for our future work.
Chapter 6

Conclusion

We presented a novel algorithm to apply semi-supervised learning to spatial knowledge transfer with deep generative models. By inferring the labels of unlabeled sensory data and using them for training, semi-supervised spatial knowledge transfer can successfully improve a deep generative model to generalize the spatial knowledge that was initially learned with a limited amount of labeled data.

Applying semi-supervised spatial knowledge transfer to mobile robots may enable lifelong learning, where an agent initially learns spatial knowledge from data which were labeled at an accessible building, then explore the real world while continuously improving its capabilities to recognize the semantics of the environment without additional supervision. Same idea can be applied to any domain of robot learning where labeled data are expensive and we want agents to keep learning from their experiences with little supervision. Our method of incremental semi-supervised learning has the potential to enable robot learning that resembles the way human infants continuously acquire new knowledge with little supervision.
Acknowledgments

The are a number of people who have supported me to complete this senior thesis and my undergraduate study, but first, I would like to express my gratitude to Dr. Andrzej Pronobis who gave me a chance to do robotics research and supervised my work throughout the project. Also, special thanks to my research partner Kaiyu Zheng for the great collaboration and inspiring discussion. I am very grateful to people who funded my study here at the University of Washington. Finally and most importantly, I want to thank my family for always loving me and encouraging me from Japan.

Kousuke Ariga
Seattle, June 2017
Bibliography


