Selecting Auxiliary Data Using Knowledge Graphs for Image Classification with Limited Labels

Elaheh Raisi
Brown University
elaheh_raisi@brown.edu

Stephen H. Bach
Brown University
stephen_bach@brown.edu

Abstract

*In this paper, we propose a learning algorithm for training deep neural networks when there is not sufficient labeled data. To improve the generalization capabilities of the deep model, we adopt a learning scheme to train two related tasks simultaneously. One is the original task (target), and the other is an auxiliary task (source). In order to create a related auxiliary task, we leverage an available knowledge graph to query for semantically related concepts that are grounded in labeled images; hence we call our method **KGAuxLearn**. We jointly train the target and source tasks in a multi-task architecture. We evaluate our method on two fine-grained visual categorization benchmarks: Oxford Flowers 102 and CUB-200-2011. Our experiments demonstrate that the error rate reduced by at least* 2.1% *over fine-tuning for both datasets. We also improve the error rate by* 1.36% *and* 2.93% *over using randomly selected concepts as an auxiliary task for Oxford Flowers 102 and CUB-200-2011, respectively. In addition, comparing our method with auxiliary data selection methods that do not use a knowledge graph, the error rate improves by* 0.69% *and* 2.57% *on Oxford Flowers 102 and CUB-200-2011, respectively.*

1. Introduction

Deep neural networks (DNNs) have been achieving state-of-the-art performances on a wide range of problems in vision, language, and speech areas [16, 21, 25]. A large part of DNNs' success is owed to the availability of massive amount of labeled data [15, 18]. However, labeling samples can be prohibitively expensive and time consuming such that it is considered one of its main bottlenecks. As one solution, there has been a growing interest in knowledge transfer methodologies such as transfer learning [28, 17, 36] and multi-task learning (MTL) [2, 42, 3]. The objective of MTL is to improve the performance of all *related* tasks while learning them in parallel. According to Caruana [8], MTL results in better performance than transfer learning because it "improves generalization by leveraging the domain-specific information contained in the training signals of related tasks." In essence, MTL introduces an inductive bias that prefers hypotheses that can explain multiple tasks, which limits overfitting [6, 5].

In this paper, we introduce a fast multi-task learning-based framework for fine-grained visual categorization, a problem which often suffers from a lack of labeled data. Previous work has demonstrated the value of selecting auxiliary data for additional tasks to cope with limited labeled data. Ge et al. [13] used expensive image-by-image similarity computations. Zhang et al. [41] used a meta learning approach that requires repeatedly training on different tasks in order to learn which auxiliary data to select. However, neither method takes advantage of high-level semantic knowledge. We show that we can use existing high-level semantic knowledge in the form of a knowledge graph to guide the selection.

In our framework, we have two tasks: target (the original learning task) and source (the auxiliary task to improve generalization on the target task). Knowledge transfer approaches such as MTL and transfer learning have better pers [8]. In order to find *related* tasks to the target data, we use ConceptNet, a semantic network of structured knowledge, which consists of more than 8 million concepts and 21 million relations [32]. In particular, our source task will be the collection of all of the retrieved concepts from the knowledge graph. For example, for the class name "rose" in Oxford Flowers 102 [27], the extracted *related* concepts include "blossom," "petal," "flower," "floribunda," "multiflora," "rose_bowl," and more. Note that there are a few non-related retrieved concepts that can be seen as noise. After extracting related concepts from the knowledge graph, we construct our source task by collecting images corresponding to those concepts from ImageNet 22K, which contains $14M$ images and $22K$ categories [10]. Then, we jointly train our deep network using both target and source tasks. Our framework architecture consists of a shared convolutional neural network followed by two parallel task-specific fully connected classifiers. Shared layers' param-

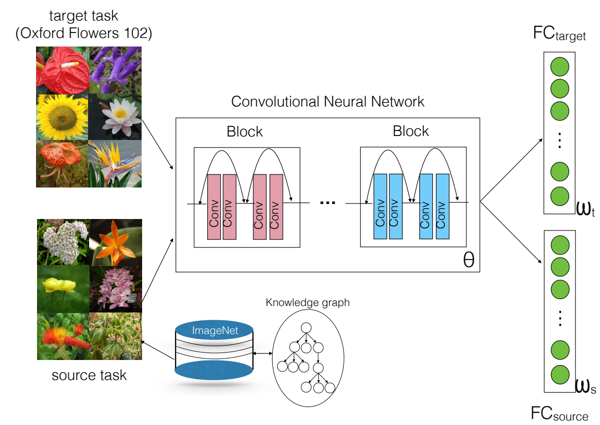

Figure 1. Illustration of *KGAuxLearn*. Inputs to the framework are the target data (such as Oxford Flowers 102) and source data (related images from ImageNet). After passing through the deep convolutional neural network, they go to the corresponding fully connected layers.

eters are updated using both target and source tasks, while each fully connected layer is updated by its corresponding task (Figure 1).

We performed our experiments on two fine-grained visual categorization (FGVC) problems, which aim to distinguish subordinate visual categories such as different species of birds [37] or flowers [27]. There are two major difficulties associated with these datasets: first, the number of training samples per class is small; and second, visual distinction between categories is subtle. As a result, training a high-quality DNN from scratch would be challenging. Our experiments on these two FGVC datasets show the efficacy of our proposed method by improving accuracy. Specifically, on Oxford Flowers 102, we decrease the error rate by 2.11%, 1.36%, and 0.69% over fine-tuning on the target task, using randomly selected concepts as the auxiliary task, and other auxiliary task selection methods, respectively. Similarly, on CUB-200-2011 we improve the error rate by 2.68%, 2.93%, and 2.57% over the aforementioned baselines.

2. Related Work

Transfer Learning In transfer learning, we leverage the knowledge we have learned from some (source) tasks to the new (target) task to combat data scarcity in addition to improving accuracy and time complexity [36, 28]. Therefore, we utilize knowledge learnt from one (or more) related task(s) to the target task. In traditional machine learning, transfer learning involves learning several interrelated problems using SVMs [38], hierarchical Bayesian methods [30], graphical models [34], Markov logic networks [24], etc. With the huge success of deep neural networks, transfer learning is fine-tuning a pretrained deep model for a target task, which improves model performance remarkably, while reducing the need for labeled data [40, 14]. In particular, in computer vision tasks, convolutional neural networks (CNNs) trained on large datasets such as ImageNet have been extensively used for transfer learning [19, 9]. For some target tasks with less labeled data such as fine-grained visual categorization, fine-tuning on deep neural networks will lead to overfitting [13]. In this paper, we try to avoid the overfitting problem by considering transfer learning, but we found that multi-task learning architecture works better on these tasks since it incorporates a regularizer in the form of *related* auxiliary task, while training the target task.

Multi-Task Learning In multi-task learning, several related tasks are learned simultaneously in order to take advantage of useful information embodied in those tasks, and increase generalization performance of all tasks [8, 36, 43]. It has shown improvements in many applications in various areas [35, 29, 11, 7, 26, 22, 39, 23, 4]. This paper is closely related to the multi-task learning paradigm with the aspect of learning more than one task. The main difference, however, is that while we train two target and source tasks simultaneously, we care about target task accuracy. The main role of the source task is to regularize the framework by learning a shared feature representation that benefits the target task the most.

Our framework is motivated by ideas studied in joint learning using auxiliary tasks [13, 41]. Ge et al. [13] used a subset of training images from the original source task for training. They computed descriptors from filter bank responses for both target and source tasks, which is computationally expensive. Zhang et al. [41] introduced a regularized meta-learning objective function in which the regularization is based on an auxiliary data. To select more useful samples from auxiliary data, they computed a score for each data point using a forward pass through their network. Then, they fine-tune the model in the second pass, which makes the framework expensive to adapt. Our framework is similar to these approaches with respect to using auxiliary data to improve the generalization performance of the model on the target task, but the main difference is the way we choose the auxiliary data. We straightforwardly use the semantic knowledge graph to retrieve all of the related concepts to the target task. This is extremely fast and easy to apply.

3. The Framework

Here is the procedure of our framework (Figure 1):

1. *Join knowledge graph to ImageNet*: First, we join the relational knowledge graph, ConceptNet [32], with the images of ImageNet 22K [10]. This enables us to query for semantically related concepts that are grounded in labeled images.

2. *Create auxiliary task*: Given the target task defined by classes that are identified with nodes in the knowledge graph, we select all of the *related* nodes to construct our source task. In order to retrieve related concepts, we use the "related" API endpoint of ConceptNet [1]. This API endpoint uses word embeddings built from a combination of ConceptNet embeddings with distributional word embeddings (word2vec and Glove) to create a more robust embedding for each concept [33]. Cosine similarity is used to find *related* terms for each node. Finally, using the connection of ConceptNet and ImageNet 22K we made in step one, we extract images to obtain auxiliary data.

3. *Construct **KGAuxLearn** architecture*: We use a deep convolutional NN followed by two task-specific fully connected layers; one for the target (FC_{target}) and the other for the source task (FC_{source}). The objective function is:

$$\underset{\boldsymbol{\theta},\boldsymbol{w_s},\boldsymbol{w_t}}{\operatorname{argmin}} \; L_{source}(h^s) + L_{target}(h^t) = \frac{1}{n}\sum_{i=1}^{n} \ell\big(h^t(\boldsymbol{x_i^t};\boldsymbol{w}_t,\boldsymbol{\theta}), y_i^t\big) + \frac{1}{m}\sum_{i=1}^{m} \ell\big(h^s(\boldsymbol{x_i^s};\boldsymbol{w}_s,\boldsymbol{\theta}), y_i^s\big)$$

where h^s and h^t are the hypotheses for *source* and *target* tasks correspondingly. $\boldsymbol{w_s}, \boldsymbol{w_t}$ are source and target parameters, and $\boldsymbol{\theta}$ is the shared parameters. The source sample set is $S_{source} = \{(\boldsymbol{x}_i^s, y_i^s)\}_{i=1}^m$, and the target sample set is $S_{target} = \{(\boldsymbol{x}_i^t, y_i^t)\}_{i=1}^n$, where m and n are the number of samples in the source and target tasks respectively.

4. *Jointly train target and source tasks*: There are three sets of parameters $\boldsymbol{w_s}, \boldsymbol{w_t}, \boldsymbol{\theta}$ that needed to be learned. In order to optimize these parameters, we applied an *alternating training procedure*, which interleaves updating source and target tasks parameters. In particular, we alternate optimizing parameter sets $\{\boldsymbol{w_s}, \boldsymbol{\theta}\}$ and $\{\boldsymbol{w_t}, \boldsymbol{\theta}\}$. Note that $\boldsymbol{\theta}$ is shared among two tasks; therefore, when the input is source data, parameter set $\{\boldsymbol{w_s}, \boldsymbol{\theta}\}$ is updated and target-related parameters FC_{target} do not change (since the gradient w.r.t these parameters is zero). When the input is target data, parameter set $\{\boldsymbol{w_t}, \boldsymbol{\theta}\}$ is updated and source-related parameters (FC_{source}) do not change.

4. Experiments

Overview We performed experiments on two fine-grained visual categorization datasets. For each target task, we queried related categories using the structural knowledge graph, ConceptNet, then extracted images of those categories from ImageNet 22K to construct the source task. Then, we adjusted a convolutional neural network by adding two task-specific classifiers on top, and fine-tuned the model using both target and auxiliary data. We computed top-1 and mean class accuracies and compared our results with baselines and other methods using auxiliary data for learning.

Datasets We evaluated our approach on two fine-grained visual categorization benchmarks as target tasks: *Oxford Flowers 102* [27] and *CUB-200-2011* [37]. Oxford Flowers 102 contains 102 classes; each class has 10 images for training and 10 images for validation. Total number of testing images is 6,149. CUB-200-2011 consists of 200 bird species with 5,994 images for training and 5,794 images for testing.

Implementation In our experiments, we use the 152-layer ResNet [15] pretrained on ImageNet 1K [31] as a deep convolutional network. The only difference is that instead of having one fully connected layer, we have two fully connected layers for source and target tasks, which are initialized randomly (Figure 1). During training, target images are augmented by a combination of random crop, flip, and rotation procedures, and during inference, images are resized and center cropped. All images are resized to 224×224. After hyperparameter tuning on the validation set, SGD with momentum 0.9 and initial learning rate 0.01 is used for CUB-200-2011 and Adam is used for Oxford Flowers 102 with initial learning rate 0.001. We applied a cosine annealing schedule to update the learning rate during training. We also freeze ResNet 152 up to layer 6. We set the maximum number of epochs to 100, and load the model which has the best *target* validation accuracy for evaluation. Our source dataset is ImageNet 22K [10], which organized according to the WordNet [12] hierarchy.

Training the Model For Oxford Flowers 102, using the "related" API endpoint of ConceptNet, we extracted 1,006 classes with $825k$ images in total, and for CUB-200-2011, we extracted 1,045 classes with approximately $885k$ training images. Note the huge difference between the number of images in source and target tasks: The source task for Oxford Flowers 102 is at least 800 times larger, and that of CUB-200-2011 is at least 160 times larger. We therefore set target task batch size to 8 and source task batch size to 128. In each iteration, we randomly select 8 samples from target and 128 samples from source task. Iterations in each epoch is based on the total number of target task, which means not all source tasks images are passed through the model in each epoch. This makes the training process very fast. In future work, we plan to use more *related* samples from the auxiliary data.

Quantitative Analysis To evaluate classification performance, we computed top-1 and mean class accuracy. We defined two baselines: (1) fine-tuning on the target task only, and (2) choosing random concepts as the source task. In the former, we just fine-tune the deep convolution network (pretrained on ImageNet 1K) with one fully connected layer for the target task. In the latter, we use the same structure as our framework, but instead of selecting a source task from the knowledge graph, we randomly sample concepts. In order to have a fair comparison, the number of randomly selected categories is exactly equal to that of the *KGAuxLearn* source task. More concretely, for Oxford Flowers 102, for example, we had retrieved 1,006 related concepts from knowledge graph to construct source task. Therefore, for the "random" baseline, we selected 1,006 random concepts to be the source task.

We also compare our method with the results of papers [13, 41] in which they used joint learning with source task from ImageNet 1K. We perform all experiments using three different seeds, and report the results in Tables 1 and 2 ($\pm$ in these tables represents standard deviation). For Oxford Flowers 102, according to Table 1, *KGAuxLearn* performance is higher than the baselines as well as *selective joint fine-tuning* introduced in [13]. In [13], their best mean accuracy was 95.8%, while we got 96.49%. Note the computational cost in their framework since they need to compute the similarities of all images descriptor in target data to those of source data, which is ImageNet 1K. For CUB-200-2011, as shown in Table 2, our model has better performance than baselines as well as *MetaFGNet* in [41]. In [41], their best accuracy (similar to our settings) is reported 80.3% when used ImageNet 1K as source data, while our model accuracy is 82.87%. Note the simplicity of our framework compared to [41], as they require one pass to compute the source task target, and the other pass for fine-tuning. For both experiments, we outperform over random auxiliary images as source task. This is significantly important as it shows our model is learning a better generalized representation while learning jointly with related tasks. It should be noted that in [41], they achieved higher accuracy when using a subset of L-Bird [20] as auxiliary set for CUB-200-2011. L-Bird contains more than $4M$ images from around $10k$ bird species. Consequently, obtaining such a high score is unsurprising when using a more specialized source task.

Few-Shot Learning Scheme As one future direction, we would like to extend our framework for few-shot learning scheme in which there are a handful number of images for each class. We performed small experiments for such setting when there are 2, 4, 6, 8, and 10 images per class for the Oxford Flowers 102 dataset. As illustrated in Figure 2, for all cases, *KGAuxLearn* outperforms all baselines. This observation is important as it shows this methodology could be fit in other related research areas such as few-shot learning.

Method	Top-1 (%)	Mean Acc (%)
Fine-tune	93.51 ± 0.14	94.38 ± 0.17
Random Source	94.35 ± 0.27	95.12 ± 0.1
Selective Joint [13]	N/A	95.8
KGAuxLearn	$\mathbf{96.07 \pm 0.20}$	$\mathbf{96.49 \pm 0.19}$

Table 1. A comparison of classification performance (top-1 and mean class accuracy with standard deviation) on Oxford Flowers 102. *KGAuxLearn* outperforms the baselines and the best result reported in [13] with ImageNet as the source task.

Method	Top-1 (%)	Mean Acc (%)
Fine-tune	80.19 ± 0.4	80.37 ± 0.38
Random Source	79.94 ± 0.3	80.11 ± 0.33
MetaFGNet [41]	80.3	N/A
KGAuxLearn	$\mathbf{82.87 \pm 0.53}$	$\mathbf{83.00 \pm 0.49}$

Table 2. A comparison of classification performance (top-1 and mean class accuracy with standard deviation) on CUB-200-2011. *KGAuxLearn* outperforms the baselines and the best result reported in [41] with ImageNet as the source task.

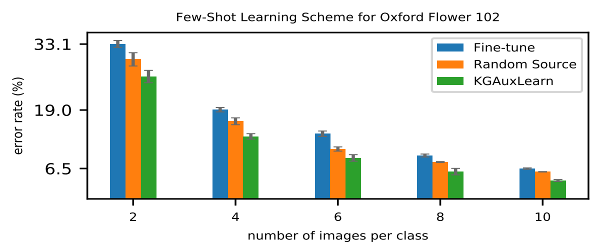

Figure 2. Error rate when there are a small number of images per class for Oxford Flowers 102 using different approaches. *KGAuxLearn* performance is better than baselines in all cases.

5. Conclusions

In this paper, we introduce a training methodology to improve image classification performance with insufficient data for training. Inspired from multi-task learning, we create a source task, related to the original task, and adjust a deep neural network to train both tasks in parallel. To construct the source task, we retrieve related concepts from a semantic knowledge graph, and then extract corresponding images in ImageNet 22K. Our experiments on two fine-grained visual categorization benchmarks are promising, illustrating accurate and fast performance.

Acknowledgments This material is based on research sponsored by Defense Advanced Research Projects Agency (DARPA) and Air Force Research Laboratory (AFRL) under agreement number FA8750-19-2-1006. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of Defense Advanced Research Projects Agency (DARPA) and Air Force Research Laboratory (AFRL) or the U.S. Government. We gratefully acknowledge the support of Google.

References

[1] Conceptnet api. https://github.com/commonsense/conceptnet5/wiki/API/. Accessed: 2019-07-12.

[2] An overview of multi-task learning in deep neural networks. https://ruder.io/multi-task/. Accessed: 2017-05-29.

[3] Andreas Argyriou, Theodoros Evgeniou, and Massimiliano Pontil. Multi-task feature learning. In *Proceedings of the 19th International Conference on Neural Information Processing Systems*, NIPS06, page 4148, Cambridge, MA, USA, 2006. MIT Press.

[4] Jing Bai, Ke Zhou, Guirong Xue, Hongyuan Zha, Gordon Sun, Belle Tseng, Zhaohui Zheng, and Yi Chang. Multi-task learning for learning to rank in web search. In *Proceedings of the 18th ACM Conference on Information and Knowledge Management*, CIKM 09, page 15491552, New York, NY, USA, 2009. Association for Computing Machinery.

[5] Jonathan Baxter. A bayesian/information theoretic model of learning to learn via multiple task sampling. In *Machine Learning*, pages 7–39, 1997.

[6] Shai Ben-David and R. Borbely. Exploiting task relatedness for mulitple task learning. pages 567–580, 01 2003.

[7] Hakan Bilen and Andrea Vedaldi. Integrated perception with recurrent multi-task neural networks. 06 2016.

[8] Rich Caruana. Multitask learning. *Machine Learning*, 28, 07 1997.

[9] Yin Cui, Yang Song, Chen Sun, Andrew Howard, and Serge J. Belongie. Large scale fine-grained categorization and domain-specific transfer learning. *CoRR*, abs/1806.06193, 2018.

[10] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. In *CVPR09*, 2009.

[11] Carl Doersch and Andrew Zisserman. Multi-task self-supervised visual learning. *CoRR*, abs/1708.07860, 2017.

[12] Ingo Feinerer and Kurt Hornik. *wordnet: WordNet Interface*, 2017. R package version 0.1-14.

[13] Weifeng Ge and Yizhou Yu. Borrowing treasures from the wealthy: Deep transfer learning through selective joint fine-tuning. *CoRR*, abs/1702.08690, 2017.

[14] Yunhui Guo, Honghui Shi, Abhishek Kumar, Kristen Grauman, Tajana Rosing, and Rogério Schmidt Feris. Spottune: Transfer learning through adaptive fine-tuning. *CoRR*, abs/1811.08737, 2018.

[15] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *CoRR*, abs/1512.03385, 2015.

[16] G. Hinton, L. Deng, D. Yu, G. E. Dahl, A. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T. N. Sainath, and B. Kingsbury. Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups. *IEEE Signal Processing Magazine*, 29(6):82–97, Nov 2012.

[17] Geoffrey Hinton, Oriol Vinyals, and Jeffrey Dean. Distilling the knowledge in a neural network. In *NIPS Deep Learning and Representation Learning Workshop*, 2015.

[18] Gao Huang, Zhuang Liu, and Kilian Q. Weinberger. Densely connected convolutional networks. *CoRR*, abs/1608.06993, 2016.

[19] Simon Kornblith, Jonathon Shlens, and Quoc V. Le. Do better imagenet models transfer better? *CoRR*, abs/1805.08974, 2018.

[20] Jonathan Krause, Benjamin Sapp, Andrew Howard, Howard Zhou, Alexander Toshev, Tom Duerig, James Philbin, and Fei-Fei Li. The unreasonable effectiveness of noisy data for fine-grained recognition. *CoRR*, abs/1511.06789, 2015.

[21] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. Imagenet classification with deep convolutional neural networks. *Commun. ACM*, 60(6):8490, May 2017.

[22] W. Liu, T. Mei, Y. Zhang, C. Che, and J. Luo. Multi-task deep visual-semantic embedding for video thumbnail selection. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3707–3715, June 2015.

[23] Minh-Thang Luong, Quoc Le, Ilya Sutskever, Oriol Vinyals, and Lukasz Kaiser. Multi-task sequence to sequence learning. *Proceedings of ICLR, San Juan, Puerto Rico*, 11 2015.

[24] Lilyana Mihalkova and Raymond Mooney. Transfer learning with markov logic networks. In *Proceedings of the ICML-06 Workshop on Structural Knowledge Transfer for Machine Learning*, Pittsburgh, PA, June 2006.

[25] Tomas Mikolov, Martin Karafit, Lukas Burget, Jan Cernock, and Sanjeev Khudanpur. Recurrent neural network based language model. volume 2, pages 1045–1048, 01 2010.

[26] Ishan Misra, Abhinav Shrivastava, Abhinav Gupta, and Martial Hebert. Cross-stitch networks for multi-task learning. *CoRR*, abs/1604.03539, 2016.

[27] M. Nilsback and A. Zisserman. Automated flower classification over a large number of classes. In *2008 Sixth Indian Conference on Computer Vision, Graphics Image Processing*, pages 722–729, Dec 2008.

[28] Sinno Pan and Qiang Yang. A survey on transfer learning. *Knowledge and Data Engineering, IEEE Transactions on*, 22:1345 – 1359, 11 2010.

[29] Anastasia Pentina and Christoph H. Lampert. Multi-task learning with labeled and unlabeled tasks. In *Proceedings of the 34th International Conference on Machine Learning - Volume 70*, ICML17, page 28072816. JMLR.org, 2017.

[30] Michael T. Rosenstein, Zvika Marx, and Leslie Pack Kaelbling. To transfer or not to transfer. 2005.

[31] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael S. Bernstein, Alexander C. Berg, and Fei-Fei Li. Imagenet large scale visual recognition challenge. *CoRR*, abs/1409.0575, 2014.

[32] Robyn Speer, Joshua Chin, and Catherine Havasi. Conceptnet 5.5: An open multilingual graph of general knowledge. *CoRR*, abs/1612.03975, 2016.

[33] Robyn Speer, Joshua Chin, and Catherine Havasi. ConceptNet 5.5: An open multilingual graph of general knowledge. pages 4444–4451, 2017.

[34] Charles Sutton and Andrew Mccallum. Composition of conditional random fields for transfer learning. 10 2005.

[35] Chen Tessler, Shahar Givony, Tom Zahavy, Daniel J. Mankowitz, and Shie Mannor. A deep hierarchical approach to lifelong learning in minecraft. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*, AAAI17, page 15531561. AAAI Press, 2017.

[36] Lisa Torrey and Jude W. Shavlik. Chapter 11 transfer learning. 2009.

[37] C. Wah, S. Branson, P. Welinder, P. Perona, and S. Belongie. The Caltech-UCSD Birds-200-2011 Dataset. Technical report, 2011.

[38] Pengcheng Wu and Thomas G. Dietterich. Improving svm accuracy by training on auxiliary data sources. In *Proceedings of the Twenty-First International Conference on Machine Learning*, ICML 04, page 110, New York, NY, USA, 2004. Association for Computing Machinery.

[39] Jianpeng Xu, Jiayu Zhou, and Pang-Ning Tan. Formula: Factorized multi-task learning for task discovery in personalized medical models. In *SDM*, 2015.

[40] Jason Yosinski, Jeff Clune, Yoshua Bengio, and Hod Lipson. How transferable are features in deep neural networks? *CoRR*, abs/1411.1792, 2014.

[41] Yabin Zhang, Hui Tang, and Kui Jia. Fine-grained visual categorization using meta-learning optimization with sample selection of auxiliary data. *CoRR*, abs/1807.10916, 2018.

[42] Yu Zhang, Ying Wei, and Qiang Yang. Learning to multitask. *CoRR*, abs/1805.07541, 2018.

[43] Yu Zhang and Qiang Yang. A survey on multi-task learning. *CoRR*, abs/1707.08114, 2017.