Hierarchical Adaptive Multi-task Learning Framework for Patient Diagnoses and Diagnostic Category Classification

Salim Malakouti
Dept. Computer Science
University of Pittsburgh
Pittsburgh, USA
salimm@cs.pitt.edu

Milos Hauskrecht
Dept. Computer Science
University of Pittsburgh
Pittsburgh, USA
milos@cs.pitt.edu

***Abstract*—The problems a patient suffers from can be summarized in terms of a list of patient diagnoses. The diagnoses are typically organized in a hierarchy (or a lattice structure) in which many different low-level diagnoses are covered by one or more diagnostic categories. An interesting machine learning problem is related to learning of a wide range of diagnostic models (at different levels of abstraction) that can automatically assign a diagnosis or a diagnostic category to a specific patient. While one can always approach this problem by learning models for each diagnostic task independently, an interesting open question is how one can leverage the knowledge of a diagnostic hierarchy to improve the classification and outperform independent diagnostic models. In this work, we study this problem by designing a new hierarchical classification learning framework in which multiple diagnostic classification targets are explicitly related via diagnostic hierarchy relations. By conducting experiments on MIMIC-III data and ICD-9 diagnosis hierarchy, we demonstrate that our framework leads to improved classification performance on individual diagnostic tasks when compared to independently learned diagnostic models. This improvement is stronger for diagnoses with a low prior and smaller number of positive training examples.**

***Index Terms*—Machine Learning, Transfer Learning, Multi-task Learning, Diagnosis Prediction, ICD-9**

I. Introduction

The widespread adoption of Electronic Health Records (EHR) in the past decade lead to emergence of new and more complex datasets covering all aspects of hospital care. These datasets provide a valuable source of information that enables the construction of variety of new models for solving more complex problems. One such problem is the problem of automatic assignment of diagnoses to a patient [1]–[4]. However, due to a low prior (and fixed dataset size) it is often impossible to learn accurate models for many of the diagnoses. The challenge is to devise methods that are more robust when facing such circumstances and that can successfully learn models for a broader range of diagnoses. In this work we explore methods that aim to leverage expert defined hierarchies and relations embedded in the hierarchies to learn improved diagnostic models.

Learning of diagnostic models from data may benefit from hierarchies in the following ways. First, low-level diagnoses in the hierarchy are abstracted to diagnostic categories. This means diagnostic categories are often easier to learn due to the fact that they come with a higher number of positive examples [5]. Second, one can often build a better diagnostic model by utilizing the model of its diagnostic parent and by learning how to differentiate it from its diagnostic siblings. Finally, one can (sometimes) also learn a model for a diagnostic category by combining results and models of its diagnostic children. In general, structuring the diagnostic decisions along diagnostic hierarchies and taking advantage of the connections among diagnoses and their categories may lead to improved models and better learning of these models [6].

In order to incorporate the aforementioned benefits of the hierarchies in learning diagnostic tasks we propose a new hierarchical adaptive learning framework that explicitly connects individual diagnostic tasks and attempts to use them to jointly learn a better collection of models. Our approach takes advantage of ideas implemented in adaptive support vector machine [7] approach and extends them to hierarchical task structures. We test our new framework on MIMIC-III data where diagnoses are defined in terms of Ninth International Classification of Diseases (ICD-9) [8] codes and their hierarchy. We show that our new framework improves upon diagnostic models built independently for each diagnostic task. We observe the effect of the hierarchy to be stronger for smaller training dataset sizes, demonstrating that our framework can leverage the presence of a hierarchical structure to compensate for lack of data and low priors when training the models.

The technical contributions of our work are two fold: (1) The design of Regularized Adaptive Support Vector Machine (RA-SVM) algorithm that can learn model parameters for a target classification task and its relation to auxiliary classification tasks simultaneously; (2) The development of a new multi-task learning framework that can leverage a predefined hierarchy of tasks to improve individual classification models by adapting parameters among parent and child tasks.

II. Related Work

Hierarchical Classification. The problem of learning a

collection of diagnostic classification models explicitly related via hierarchy is referred to in the machine learning literature as hierarchical classification [9]. The most common method for defining classification models within a hierarchy is to use the top-down approach [10]. In this case, a classifier on a low-level of the hierarchy is defined using a decision or the signal generated by its parent classifiers. There are different versions of the top down approach that place various consistency constraints on predictions of the parent and child tasks and their classifier outputs, most frequently assuring the probability of a parent (diagnostic category) is higher than the probability of a low-level class or class category [11], [12]. The main problem with the top down approach is that learning of higher level class models from data may omit details only low-level class models can capture. For example, some of the findings for a patient may point specifically and with a high accuracy to a low-level diagnosis while the higher level class model marginalizes it out during the learning and as a result does not include it in the model. In such a case the probability of a lower-level class may be higher than the probability of a higher level class category violating the constraint consistency. One way to correct for child-to-parent effects is to define and add a bottom-up process that assures positive lower-level class predictions aggregate properly in the parent tasks [13]. However, pure bottom-up approach would require the presence of accurate classifier models on the leaf classification layer, which is hard to achieve in practice when datasets of a limited size are used to train such models and the count of positive instances for such classes are very low. There exists a variety of hierarchical classification methods that try to account for both the top-down and bottom-up classification processes. One example is a Bayesian aggregation method by [14] that compiles the hierarchy into the Bayesian belief network and uses inferences to support the classification on a different level of hierarchy. The limitation of the vast majority of current methods is that classification models are dependent to related models both during the learning and the application stage. One advantage of our framework is that while it considers the model interactions during the training stage, it leads to separate models that can be applied independently.

Multi-task Learning. Multi-task learning refers to a category of machine learning methods that learn multiple related tasks simultaneously. The motivation is to exploit task relationships, commonalities and differences. This has shown promising results in improving individual tasks compared to the standard solution of learning each task independently [15]. Evgeniou and Pontil proposed a method based on Support Vector Machines (SVM) algorithm that learns all tasks simultaneously by regularizing their differences from their average [16]. Argyriou et al. proposed a multi-task feature learning method that attempts to learn a lower dimensional feature space that is shared across all tasks [17]. However, when tasks in multi-task settings are not related or similar negative transfer can happen [18]. Existing work that have tried to tackle this problem fit in two categories. The first group tackles negative transfer problem by learning task relationships [19]. The second category of solutions attempts to learn task clusters to prevent negative transfer from unrelated tasks [20]. A shortcoming of the many existing work in the later group is that they have assumed tasks reside in a flat cluster structure and do not consider hierarchical structures. Recent work have attempted to take advantage of hierarchies by imposing regularization on tasks based on groups on different levels of the hierarchy, assuming that target tasks only reside in the leaf nodes of the hierarchy [21], [22]. However, in our problem, target tasks are nodes in any level of the hierarchy. Additionally, they do not consider the transfer of weights directly from parents to children and vice-versa. In our method, not only we consider the potential top-down and bottom-up transfer of weights based on a given hierarchy, but we also allow models to learn the usefulness of both their parents and children to prevent negative transfer.

Patient diagnosis prediction and classification. The majority of existing work on modelling patient diagnosis attempts to either (1) predict patient's diagnoses for the next patient visit, or, (2) to assign diagnoses for the current visit at the time of patient discharge with the goal of improving hospitals' billing process [1], [2]. Lipton et al. proposed a Recurrent Neural Network (RNN) architecture based on Long Short Term Memory (LSTM) units to predict future patient visits' diagnosis from a collection of 13 clinical variables [3]. GRAM [4] is an attention based RNN that uses a bag-of-word (BoW) representation of patient's previous diagnoses as their input and takes advantage of diagnosis hierarchies to extend low level diagnoses tasks by adding diagnostic categories.

The problem studied in this paper is closer to the second problem. However, the main difference from the past work is that we actively use hierarchical relations in the diagnostic hierarchy and multi-task learning techniques to learn better models. Models that can take advantage of any existing unsupervised method for learning dense representation of patient's Electronic Health Record (EHR) data as features.

III. Proposed Methodology

Our goal is to learn predictive models for T tasks corresponding to diagnoses and diagnostic categories organized in a hierarchy. Each individual diagnostic task maps a dense representation of information in patient's EHR (X) to one of the $\{0, 1\}$ labels. Each label reflects whether a specific diagnosis or a diagnostic category should be assigned to the patient defined by the information in X. The specifics of the X representation used in this paper will be covered in the experiments section since this is not a main focus of our work. We assume T diagnostic models are defined with the help of discriminant projections $f_1, f_2, ..., f_T$, $f_t : X \rightarrow R$ where the specific class assigned to X for the task t depends on a threshold α_t defined on possible values of f_t.

A conventional approach is to learn each projection f_t independently. However, multi-task learning literature has shown that simultaneous learning of tasks can improve model performances [15], [18]. Unfortunately, in scenarios in which

a large number of heterogeneous tasks exist, many multi-task learning algorithms that do not incorporate task relationships face negative transfer [18]. Hence, other multitask learning methods have been proposed to learn relationships of target tasks to ultimately prevent negative transfer [19], [23].

Our objective in this work is to use a diagnostic hierarchy to guide the transfer of model parameters. Intuitively, when learning a diagnostic model, one can benefit from utilizing the models both from its immediate parent and children. This idea leads to the following diagnostic model for task t:

$$\begin{aligned} f_t(x) = \Sigma_{j \in parent(t)} \tau_j f_j(x) \\ + \Sigma_{i \in child(t)} \tau_i f_i(x) + \Delta f_t(x) \end{aligned} \tag{1}$$

where parameters τ_k reflect the amount of transfer from task k and $\Delta f_t(x)$ is the task specific component formed by a linear combination of features in x. Learning of the best set of parameters $\Delta f_t(x)$ and transfer parameters τ_k is tricky because of circular dependencies in the definition of the functions. In this work we solve the above problem by defining a two step (pass) algorithm to transfer parameters from one task to another. First, our algorithm learns a set of models by following the hierarchy in top-down fashion where the transfer proceeds from higher-level diagnostic categories to lower-level diagnoses. Second, it uses the hierarchy to transfer the info in the bottom-up pass by adapting the model parameters from lower level diagnoses to their immediate parents.

More formally, in the first top-down pass we learn models:

$$f_t^{td}(x) = \Sigma_{j \in parent(t)} \tau_j f_j^{td}(x) + \Delta f_t(x) \tag{2}$$

that ignore the influences from children. In the bottom-up pass we consider the influences from children models:

$$\begin{aligned} f_t^{bu}(x) = \tau_{td} f_i^{td}(x) \\ + \Sigma_{i \in child(t)} \tau_i f_i^{bu}(x) + \Delta f_t(x) \end{aligned} \tag{3}$$

Please note that both set of parameters τ_i and $\Delta f_t(x)$ are re-optimized in every pass. The term $\tau_{td} f_i^{td}(x)$ in (3) represents a self adaption mechanism that enables transfer of parameters from the previous version of f_t trained to allow the model to keep any positive improvement during the top-down pass.

The above process consists of learning a set of models $f_1, f_2, ..., f_T$ by adapting model parameters from hierarchically related or auxiliary models. Let $aux(t)$ define a set of auxiliary models for model t used to train a specific version of f_t. We can rewrite the models trained in each pass as:

$$f_t(x) = \Sigma_{i \in aux(t)} \tau_i f_i(x) + \Delta f_t(x) \tag{4}$$

by simply varying the models included in the $aux(t)$ set. To present our learning solution, we first review Adaptive Support Vector Machines (A-SVM) algorithm [7]. A-SVM allows adaptive learning of $f_t(x)$ from the auxiliary tasks. However, A-SVM assumes that weight of auxiliary tasks are known in advance. Hence, we propose Regularized Adaptive Support Vector Machine (RA-SVM) as a variation of A-SVM that simultaneously learns τ_a values and model parameters.

A. Adaptive Support Vector Machine

Adaptive Support Vector Machine is a transfer learning algorithm that learns a function f_t for a target task t by taking advantage of pre-trained models for a set of auxiliary tasks. The idea, first proposed in [7] is to learn a set of parameters w_t for target task t by adapting and tuning a set of given model parameters for related auxiliary tasks. A-SVM learns a new function Δf_t to predict how much the predictions for target task t should differ from the predicted scores of its auxiliary tasks. Therefore, it defines $f_t = \Sigma_{a \in A} \tau_a f_a + \Delta f_t$ in which τ_a determines the contribution of an auxiliary task a while $\Sigma_{a \in aux(t)} \tau_a = 1$. The generalized version of A-SVM for multiple auxiliary tasks can be formulated as shown in (5).

$$\begin{aligned} \min_{v_t, \varepsilon} \quad & \Sigma_i^{N_t} \varepsilon_i + C||v_t|| \\ s.t. \quad & y_i \Sigma_a \tau_a f^a(x_i) + y_i v_t^T x_i \geq 1 - \varepsilon_i \\ & i = 1, ..., N, \quad \varepsilon_i \geq 0 \end{aligned} \tag{5}$$

In (5), $\Delta f_t = v_t x_i^T$ and C determines the balance between minimizing regularization term $||v_t||$ and the loss function. Larger values of C result in stronger regularization of model parameters which forces more similarities between f_t and $\Sigma_a \tau_a f_a(x_i)$. On the other hand, smaller values of C allow f_t to differ from its auxiliary models. Although A-SVM is able to use any arbitrary auxiliary model as input, if all f^a functions are linear SVM models one can calculate $w_t = \Sigma_{a \in aux(t)} \tau_a w_a + v_t$. This allows us to use task t's model independently from its auxiliary models.

B. Regularized Adaptive Support Vector Machines

One shortcoming of A-SVM is that it requires the impact weight of each auxiliary task a as τ_a to be determined beforehand. This, however, is not sufficient for learning of large hierarchies of tasks. Instead, it is favorable to use an algorithm that can simultaneously learn importance of each auxiliary task. Therefore, we propose Regularized Adaptive SVM (RA-SVM), a new version of A-SVM which simultaneously learns the usefulness of auxiliary tasks while learning model parameters v_t (Δf_t). We achieve this by relaxing the assumption $\Sigma_{a \in aux(t)} \tau_a = 1$. Thus, as shown in (6), we introduce a new regularization term to regularize $\tau = [\tau_1, ..., \tau_a, ..., \tau_{|aux(t)|}]$ in which τ_a is the influence of auxiliary task a.

$$\begin{aligned} \min_{v_t, \varepsilon, \tau} \quad & \Sigma_i^{N_t} \varepsilon_i + C_1||v_t||^2 + C_2||\tau||^2 \\ s.t. \quad & y_i \Sigma_a \tau_a f^a(x_i) + y_i v_t^T x_i \geq 1 - \varepsilon_i \\ & i = 1, ..., N_t, \quad \varepsilon_i \geq 0 \end{aligned} \tag{6}$$

Values of C_1 and C_2 determine the trade-off between regularizing model parameters and auxiliary task weights. We defined λ as $\lambda = \frac{C_2}{C_1}$. Higher values of λ will push further regularization of τ and therefore increase the impact of v_t in determining f. This translates to our tendency to independently learn the model parameters for task t. On the other hand, smaller values of λ imply that we prefer the model for task t to be more similar to auxiliary task models.

While the value of λ still needs to be determined using cross-validation or prior knowledge, it decreases the search space significantly while having an intuitive interpretation.

Because of the simplicity of RA-SVM, we can perform the minimization problem in (6) by converting it to a standard SVM optimization problem. To do so, we define $F(x_i) = [f_1(x_i), ..., f_{|aux(t)|}(x_i)]$ for all auxiliary tasks. Next, we define a new weight vector v' and feature map ϕ over input feature x as shown in (7). Additionally, we define the cost parameter (C) in the standard SVM $C = \frac{1}{2C_1}$ and μ as $\mu = \sqrt{\lambda}$ for L2 regularization and $\mu = \lambda$ if L1 regularization is used. Therefore, the optimization problem in (6) can be re-written using new parameters and input shown in (7) and hence solved using any standard SVM library.

$$v' = [v_t, \mu\tau], \quad \phi(x_i) = [x_i, \frac{1}{\mu}F(x_i)] \tag{7}$$

IV. Experiments

In this section we first describe the the used dataset, the adopted method for obtaining dense representation of patient's EHR data and evaluation metrics. Finally, we provide quantitative results and qualitative analysis of our method.

A. Dataset

We conducted our experiments on MetaVision subset of MIMIC-III dataset [24] which consists of 22046 patient visits. The diagnoses in MIMIC-III are recorded using ICD-9 codes. Therefore, we rely on ICD-9 hierarchy to obtain diagnostic categories. Categorical ground truth labels are obtained by applying a logical OR operation between all its children. Furthermore, we limited the list of diagnostic tasks to a subset of 696 codes with a minimum prior of 0.01 for positive class to ensure that enough positive samples are available to perform internal sub-sampling for hyper parameters optimization.

B. Learning Dense Representation of Patient's EHR Data

The problem of obtaining dense representation of patient's EHR data has recently been studied rigorously. Various solutions based on Singular Vector Decomposition (SVD) [5], Non-negative Matrix Factorization [25] and deep learning [2], [26], [27] have been proposed in past. DeepPatient, for example, takes advantage of auto-encoder networks to obtain such dense representations [2].

In this work, we utilized Latent Semantic Indexing (LSI) [28]. LSI uses Support Vector Decomposition to learn a lower dimensional representation of original data. This allows us to find a task-independent representation of patient's EHR data. In order to use LSI, we first converted information in patient's EHR data to a set of meaningful binary events (words). We converted medication and procedure orders to occurrence indicators. Laboratory results and physiological measurements with numerical values were converted to Normal and Abnormal Low or High events based on their standard normal ranges. Discrete valued measurements and pain level assessments were also converted to specific events matching each unique value. After the conversion, our new EHR events data consisted of 4826 clinical events including 2420 for medication orders, 116 for procedure orders, 2012 for laboratory results and 278 for physiological and pain assessment measurements. Finally, we created a BoW representation of patient events data with normalized frequencies. This results in a patient-event matrix $E_{N\times Q}$ in which N is the number of patient visits and Q represents the total number of clinical events.

TABLE I
Average performance for all diagnostic tasks across different data sizes

Method Name	AUROC	AUPRC
Random (N=500)	0.5	0.065
SVM (N=500)	0.636	0.124
HA-MTL (N=500)	**0.656**	**0.13**
HA-MTL$_{td}$ (N=500)	0.655	0.129
Random (N=1000)	0.5	0.065
SVM (N=1000)	0.671	0.144
HA-MTL (N=1000)	**0.694**	**0.15**
HA-MTL$_{td}$ (N=1000)	0.692	0.148
Random (N=5000)	0.5	0.065
SVM (N=5000)	0.739	0.181
HA-MTL (N=5000)	**0.751**	**0.185**
HA-MTL$_{td}$ (N=5000)	0.746	0.184

C. Quantitative Results

In this section we provide comparison of our model's performance to the following baselines:

- **Random**: A random guessing baseline
- **SVM**: SVM models trained independently for each task
- **HA-MTL**: The complete version of our proposed method
- **HA-MTL**$_{td}$: Top-down only version of HA-MTL

In order to compare the performance of our method with baselines, we used Area Under Receiver Operating Curve (AUROC) and Area Under Precision Recall Curve (AUPRC). Finally, in order to test the significance of the improvements by our method, we used Wilcoxon signed-rank test which has been shown to be a more suitable test for comparing performance of two classifiers on multiple datasets [29]. Additionally, we used random sub-sampling to generate 10 different 75%/25% train/test splits to evaluate the performance of the four methods described above. Moreover, we use 5 rounds of internal random sub-sampling for hyper parameters optimization using the training set.

Table I shows the average AUROC and AUPRC of all tasks for HA-MTL and HA-MTL$_{td}$ compared to baselines. Our method is outperforming the baselines in average and across different training sizes. This shows that our method has been able to effectively find the useful auxiliary tasks and adapt model parameters in even lower training data sizes. However, it seems that the majority of improvements happen during the top-down step. This is also further shown in section IV-D where we study the transfer weights of auxiliary tasks. Moreover, the difference with SVM results was found statistically significant ($p\_value < 0.05$)

We don't expect to improve all diagnostic models, instead, we expect to improve weaker models that can be improved by transferring model weights from their parents and child tasks.

Therefore, the impact of HA-MTL is more clear if we study improvements in specific diagnostic models. Table II depicts significant model performance improvements for weaker and lower level diagnostic tasks across multiple branches of the hierarchy. This conclusion is further confirmed by studying the model weights in section IV-D.

TABLE II
COMPARISON OF METHODS FOR EXAMPLE BRANCHES OF ICD9

Diagnostic Task Name	**SVM AUROC**	**HA-MTL AUROC**
Heart failure (N=500)	0.846	0.849
_ Systolic heart failure	0.797	0.801
___ Acute systolic heart failure	0.785	0.821
_ Combined systolic and diastolic heart failure	0.777	0.84
___ Acute/chronic syst/dias heart failure	0.799	0.882
Diabetes mellitus	0.88	0.885
_ Diabetes mellitus withoout complications	0.773	0.792
___ Type 2 diabetes mellitus wo complications	0.711	0.773
_ Diabetes with ophthalmic manifestations	0.833	0.872
___ Type 2 diabetes w ophthalmic manifestation	0.731	0.846

Similar trends as in "Heart Rate" and "Diabetes mellitus" can be observed in various branches of the ICD9 hierarchy: HA-MTL improved performance of individual diagnoses and diagnostic categories up to 12.6% in AUROC and 20.6% in AUPRC for some tasks. In general more than 24% of tasks were improved at least by 5% in AUROC. This is while very small number of tasks were impacted by negative transfer. Overall near 85% of tasks either improved or maintained their performance while only 0.05%(4 out of 698) of the target tasks faced a decrease in AUROC close to or greater than 5%.

D. Learned Auxiliary Task Weights

Fig. 1 (a) and (b) show the learned transfer weights for the top-down and bottom-up steps[1]. We see that transfer of parameters occurs in both steps but parent diagnoses have stronger impact on improving child diagnoses. This agrees with our quantitative results showing the top-down step's impact is more significant. This can be explained by two intuitive reasons: First, diagnostic categories have a higher number of positive samples. Second, diagnostic categories, if defined properly, represent more general diagnostic tasks that are easier for training a model as shown in past work [5]. Therefore, stronger models of parent diagnostic categories can translate into higher impacts in the top-down step.

Fig. 1 (c) and (d) illustrate the weights of auxiliary tasks for top-down and bottom-up steps for the "Heart Failure" branch. We saw earlier in Table II that tasks under "Heart failure" are improved by the top-down step while "Heart failure" itself was not significantly improved. This can also be seen in Fig. 1 (c) and (d) that parents generally have a higher impact on the children. This impact is as high as 0.96 for adaption of parameters from the diagnostic category "Combined systolic and diastolic heart failure" to "Ac/chr syst/diast heart failure", which means the parent model has equal importance as the learned model parameters for the target task (See (6)).

[1]An interactive version is available at http://cs.pitt.edu/~salimm/hamtl/

V. CONCLUSION

We proposed a hierarchical adaptive multi-task learning framework for learning classification models for patient diagnoses and diagnostic categories. Our method learns diagnostic models through a two step process. First, it performs a top-down step that transfers model parameters from parents to children. Second, it performs a bottom-up pass that learns improved parent models by adapting from their children. By conducting experiments on MIMIC-III data and ICD-9 diagnosis hierarchy, we have demonstrated that our framework leads to improved performance when compared to independently learned models. This improvement is stronger for diagnoses with a low prior and well-defined parent categories.

Acknowledgement. This work was supported by NIH grant R01-GM088224. The content of this paper is solely the responsibility of the authors and does not necessarily represent the official views of the NIH.

REFERENCES

[1] S. V. Pakhomov, J. D. Buntrock, and C. G. Chute, "Automating the assignment of diagnosis codes to patient encounters using example-based and machine learning techniques," *Journal of the American Medical Informatics Association*, vol. 13, no. 5, pp. 516–525, 2006.

[2] R. Miotto, L. Li, B. A. Kidd, and J. T. Dudley, "Deep patient: an unsupervised representation to predict the future of patients from the electronic health records," *Scientific reports*, vol. 6, p. 26094, 2016.

[3] Z. C. Lipton, D. C. Kale, C. Elkan, and R. Wetzel, "Learning to diagnose with lstm recurrent neural networks," *arXiv preprint*, 2015.

[4] E. Choi, M. T. Bahadori, L. Song, W. F. Stewart, and J. Sun, "Gram: graph-based attention model for healthcare representation learning," in *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 2017, pp. 787–795.

[5] S. Malakouti and M. Hauskrecht, "Predicting patients diagnoses and diagnostic categories from clinical-events in ehr data," in *Conference on Artificial Intelligence in Medicine in Europe*. Springer, 2019, pp. 125–130.

[6] A. Perotte, R. Pivovarov, K. Natarajan, N. Weiskopf, F. Wood, and N. Elhadad, "Diagnosis code assignment: models and evaluation metrics," *Journal of the American Medical Informatics Association*, vol. 21, no. 2, pp. 231–237, 2013.

[7] J. Yang, R. Yan, and A. G. Hauptmann, "Cross-domain video concept detection using adaptive svms," in *Proceedings of the 15th ACM international conference on Multimedia*. ACM, 2007, pp. 188–197.

[8] V. N. Slee, "The international classification of diseases: ninth revision (icd-9)," *Annals of internal medicine*, vol. 88, no. 3, pp. 424–426, 1978.

[9] C. Silla and A. Freitas, "A survey of hierarchical classification across different application domains," *Data Mining and Knowledge Discovery*, vol. 22, pp. 31–72, 01 2011.

[10] D. Koller and M. Sahami, "Hierarchically classifying documents using very few words," in *Proceedings of the Fourteenth International Conference on Machine Learning*, ser. ICML '97. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1997, pp. 170–178.

[11] S. Dumais and H. Chen, "Hierarchical classification of web content," in *Proceedings of the 23rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, ser. SIGIR '00. New York, NY, USA: ACM, 2000, pp. 256–263.

[12] F. Wu, J. Zhang, and V. Honavar, "Learning classifiers using hierarchically structured class taxonomies," in *Lecture Notes in Computer Science*, vol. 3607. Springer, Germany, 2005.

[13] G. Valentini, "True path rule hierarchical ensembles for genome-wide gene function prediction," *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, vol. 8, no. 3, pp. 832–847, May 2011.

[14] C. DeCoro and Z. Barutcuoglu, "Hierarchical shape classification using bayesian aggregation," in *IEEE International Conference on Shape Modeling and Applications 2006(SMI)*, vol. 00, 06 2006, p. 44.

[15] Y. Zhang and Q. Yang, "A survey on multi-task learning," *arXiv preprint arXiv:1707.08114*, 2017.

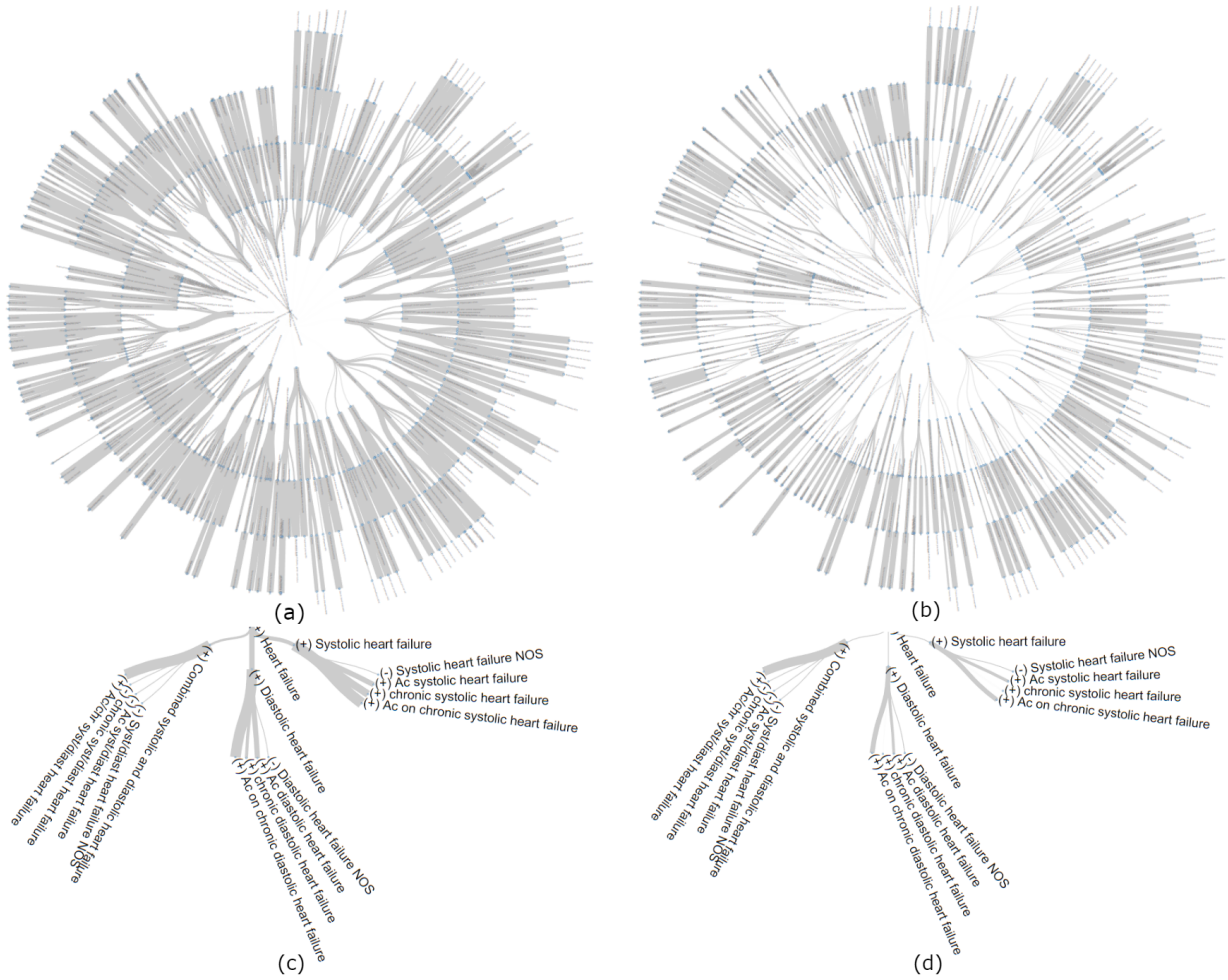

Fig. 1. Figures (a)/(c) and (b)/(d) show weights of top-down and bottom-up steps for the entire hierarchy and the subset of it that belongs to "Heart Failure". Wider edges indicate higher weights and higher impact of auxiliary tasks. (+)/(-) signs show if a model was trained for the diagnosis code (see section IV-A).

[16] T. Evgeniou and M. Pontil, "Regularized multi–task learning," in *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2004, pp. 109–117.

[17] A. Argyriou, T. Evgeniou, and M. Pontil, "Multi-task feature learning," in *Advances in neural information processing systems*, 2007, pp. 41–48.

[18] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Transactions on knowledge and data engineering*, vol. 22, no. 10, pp. 1345–1359, 2010.

[19] S. Ben-David and R. Schuller, "Exploiting task relatedness for multiple task learning," in *Learning Theory and Kernel Machines*. Springer, 2003, pp. 567–580.

[20] L. Jacob, J.-p. Vert, and F. R. Bach, "Clustered multi-task learning: A convex formulation," in *Advances in neural information processing systems*, 2009, pp. 745–752.

[21] A.-A. Liu, Y.-T. Su, W.-Z. Nie, and M. Kankanhalli, "Hierarchical clustering multi-task learning for joint human action grouping and recognition," *IEEE transactions on pattern analysis and machine intelligence*, vol. 39, no. 1, pp. 102–114, 2017.

[22] S. Kim and E. P. Xing, "Tree-guided group lasso for multi-task regression with structured sparsity." in *ICML*, vol. 2, 2010, p. 1.

[23] Z. Kang, K. Grauman, and F. Sha, "Learning with whom to share in multi-task feature learning." in *ICML*, vol. 2, no. 3, 2011, p. 4.

[24] A. E. Johnson, T. J. Pollard, L. Shen, H. L. Li-wei, M. Feng, M. Ghassemi, B. Moody, P. Szolovits, L. A. Celi, and R. G. Mark, "Mimic-iii, a freely accessible critical care database," *Scientific data*, vol. 3, p. 160035, 2016.

[25] J. C. Ho, J. Ghosh, and J. Sun, "Marble: high-throughput phenotyping from electronic health records via sparse nonnegative tensor factorization," in *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2014, pp. 115–124.

[26] E. Choi, M. T. Bahadori, E. Searles, C. Coffey, M. Thompson, J. Bost, J. Tejedor-Sojo, and J. Sun, "Multi-layer representation learning for medical concepts," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 2016, pp. 1495–1504.

[27] J. M. Lee and M. Hauskrecht, "Recent context-aware lstm for clinical event time-series prediction," in *Conference on Artificial Intelligence in Medicine in Europe*. Springer, 2019, pp. 13–23.

[28] S. Deerwester, S. T. Dumais, G. W. Furnas, T. K. Landauer, and R. Harshman, "Indexing by latent semantic analysis," *Journal of the American society for information science*, vol. 41, no. 6, pp. 391–407, 1990.

[29] J. Demšar, "Statistical comparisons of classifiers over multiple data sets," *Journal of Machine learning research*, vol. 7, no. Jan, pp. 1–30, 2006.