Structured Knowledge Graphs for Classifying Unseen Patterns in Radiographs

Chinmay Prabhakar[1] CHINMAY.PRABHAKAR@UZH.CH
Anjany Sekuboyina[1,2] ANJANY.SEKUBOYINA@UZH.CH
Hongwei Bran Li[1,2] HONGWEI.LI@TUM.DE
Johannes C. Paetzold[2] JOHANNES.PAETZOLD@TUM.DE
Suprosanna Shit[1,2] SUPROSANNA.SHIT@TUM.DE
Tamaz Amiranashvili[1,2] TAMAZ.AMIRANASHVILI@UZH.CH
Jens Kleesiek[3] JENS.KLEESIEK@UK-ESSEN.DE
Bjoern Menze[1] BJOERN.MENZE@UZH.CH
[1] *Department of Quantitative Biomedicine, University of Zurich*
[2] *Department of Computer Science, Technical University of Munich*
[3] *Institute for AI in Medicine (IKIM), University Hospital Essen*

Abstract

The presence of annotated datasets is crucial to the performance of modern machine learning algorithms. However, obtaining richly annotated datasets is not always possible, especially for novel or rare diseases. This becomes especially challenging in the realm of multi-label classification of chest radiographs, due to the presence of numerous unknown disease types and the limited information inherent to x-ray images. Ideally, we would like to develop models that can reliably label such unseen patterns (classes). In this work, we present a knowledge graph-based approach to predict such novel, unseen classes. Our method directly injects the semantic relationships between seen and unseen disease classes. Specifically, we propose a principled approach to parsing and processing a knowledge graph conditioned on the given task. We show that our method matches the labeling performance of the state-of-the-art while outperforming it on unseen classes by a substantial **2**% gain on chest X-ray classification. Crucially, we demonstrate that embedding disease-specific knowledge as a graph provides inherent explainability. (The code is available at https://github.com/chinmay5/ml-cxr-gzsl-kg)

Keywords: generalized zero-shot learning, knowledge graphs, graph neural networks

1. Introduction

In recent years, deep learning-based computer-aided diagnostic systems have achieved expert-level performance in some challenging tasks (Rajpurkar et al., 2017; Esteva et al., 2017; De Fauw et al., 2018). However, existing methods typically rely on large-scale fully annotated datasets, are often single-modal and are limited to the concepts visible during training. Such limitations magnify in the scenario of novel and rare diseases. This is especially the case in multi-label x-ray image classification tasks where multiple diagnoses (labels) per image exist. It is infeasible to collect sufficient paired image and annotation for every possible combination of disease types during training. Consequently, existing systems are limited by the expressivity of their training annotations and are unable to predict unseen diseases. However, holistic predictions are essential to facilitate optimal clinical treatment.

© 2022 C. Prabhakar[1] *et al.*

Therefore clinicians usually integrate diverse information (e.g., literature, prior experience, symptomatic correlations, etc.) to recognize novel unseen diseases.

Generalized zero-shot learning aims to address this issue of annotation scarcity. The models are trained to classify certain diseases (i.e., seen classes). During inference, they are expected to *also* classify unobserved diseases. In other words, the models are expected to perform well at classifying *new* diseases while retaining their performance on the ones already encountered during the training. One critical step in building models that can work well on both seen and unseen classes, is to incorporate 'clinical knowledge' to establish a relation between the seen and unseen diseases.

Existing methods employ natural language models e.g. *Word2Vec* (Goldberg and Levy, 2014; Zhang et al., 2019), *BERT* (Devlin et al., 2018), or the domain-specific *BioBERT* (Alsentzer et al., 2019) to bridge this information gap. However, such word embedding models are trained using a word co-occurrence objective and do not always explicitly encode knowledge in a clinical setting (Schick and Schütze, 2020). As an alternative, we propose to exploit a more explicit knowledge representation in the form of various knowledge graphs of medical ontologies. Such knowledge graphs consist of millions of medical entities (e.g., diseases, anatomical locations, medicines etc.) and the relationships between them. However, expressive knowledge graphs in medical domain often constitute an ultra large database and lack efficient ways of parsing and processing. The size of such knowledge graphs grow exponentially with the amount of ontological granularity they hold, e.g. the Unified Medical Language System (*UMLS*) (Bodenreider, 2004). Thus, their efficient usage becomes challenging in practice. Our work aims to efficiently incorporate such medical KG as a source of rich semantic knowledge.

In this work, we attempt to classify *multi-label* chest x-rays in a generalized zero shot learning setting. We build a large knowledge graph from *UMLS* to bridge the semantic information gap. Specifically, our contributions are three-folds:

1. We are the first to propose the usage of *UMLS* as a source of semantic information in the GZSL setup. We utilize the parsed knowledge from the UMLS for multi-label disease classification in chest x-rays. We improve upon state-of-the-art methods by a substantial 2% gain.
2. We validate our approach to two chest x-ray datasets with non-identical disease labels, thus confirming the generalizability of our proposed method.
3. Since incorporating semantic knowledge as a graph offers inherent explainability, we explore to use the *GNNExplainer* (Ying et al., 2019) to draw medical intuitions.

2. Related Work

Generalized zero-shot learning with knowledge graphs. In the natural image domain, knowledge graphs can effectively bridge the semantic gap between seen and unseen classes (Wang et al., 2018; Zhao et al., 2017; Xian et al., 2017; Li et al., 2020). The graphs are constructed with nodes representing individual classes and edges indicating a semantic relation between these classes. In the medical domain, Chen et al. (2020a) proposed to use label co-occurrences that appeared in the training set to generate a knowledge graph. However, this approach is not applicable in the generalized zero shot learning setting since

the unseen co-occurrences are not a part of the graph. Instead, we turn to knowledge graphs (KG). They are semantically rich and contain relationships between a vast range of medical concepts. Thus, we use KG to construct semantically rich graphs and extend its applicability to different diagnosis tasks.

Generalized zero-shot learning for multi-label tasks. In the multi-label setting, the generalized zero shot learning aims to classify a given image associated with multiple labels, a setup relatively unexplored in chest radiographs. Paul et al. (2021) propose a trait-guided multi-view semantic embedding strategy but assumes the availability of radiology reports along with the radiographs. Hayat et al. (2021) propose to create an end-to-end network that jointly learns visual representations from radiographs and aligns them to the semantic features by using BioBERT embeddings (Alsentzer et al., 2019). The method aligns the visual features with their semantic label embeddings. In contrast, we show that the relational clinical information from KG can be a better embedding than using only BioBERT.

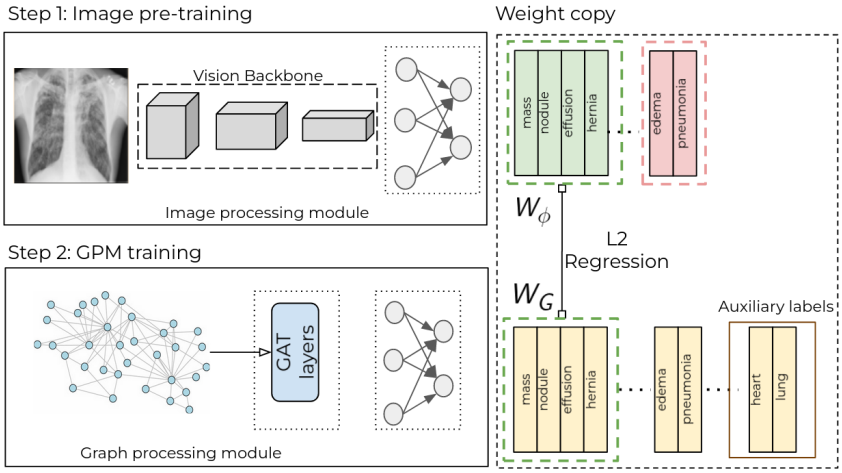

Figure 1: The proposed training pipeline. First, the vision backbone is trained with samples of the *seen* classes. This generates *Visual Classifier Weights* W_ϕ for each of the target labels. In the second step, the Graph Processing Module (*GPM*) is trained using a normalized L2 regression loss (Eq. 2) between the *Visual Classifier Weights* and weights learned by final layer of *GPM* (referred to as *GPM Weights* W_G) using only the *seen* class' weights. In the final step, the *GPM* weights W_G replace the classification head of the Image processing module. We fix these *GPM* weights and fine tune the image processing module.

3. Method

Overview. Consider a multi-label set $\mathcal{Y}$ consisting of a total of C classes. Of these C classes, only S are seen during training, and U classes are unseen. Let $\mathcal{Y}^S$ and $\mathcal{Y}^U$ denote the label sets for the seen and unseen classes, respectively. Thus, $\mathcal{Y}^C = \mathcal{Y}^S \cup \mathcal{Y}^U$, where $\mathcal{Y}^S = \{y_1, y_2, ..., y_S\}$ and $\mathcal{Y}^U = \{y_{S+1}, y_{S+2}, ..., y_C\}$. Note that $\mathcal{Y}^S \cap \mathcal{Y}^U = \emptyset$ i.e. training images contain only seen labels. The label vector $y_i \in \{0, 1\}^S$ indicates the presence of every

seen class. During training, images containing only the seen labels $\mathcal{Y}^S$ are given. During inference, given an image x_{test}, the model is supposed to correctly predict the labels from both seen and/or unseen classes.

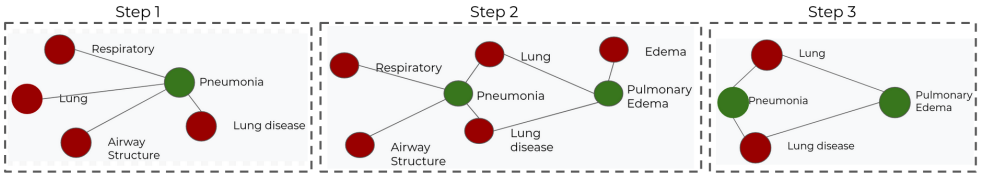

Figure 2: Parse logic of *UMLS* for a 1-hop neighbourhood. The target labels (in green) act as *seed* points. In the first step, a target label is chosen at random, and its directly connected relations are extracted from the *UMLS*. This might produce entities not part of the target labels (in red). Next, the same process is repeated for the remaining target labels. In the final step, we prune the resulting graph by retaining the nodes and edges that are part of *All Pair Shortest Path* with respect to the target nodes.

Proposed architecture. Our proposed solution consists of two main components, a Graph processing module (*GPM*) and an image processing module. The *GPM* is responsible for processing the knowledge graph (*KG*) and generating node features for the disease labels. The *GPM* is realized using a series of graph convolution layers (Brody et al., 2021). The image processing module is responsible for processing the input chest radiographs and is a DenseNet-121 backbone (Huang et al., 2018).

We train the model in two distinct steps (Figure 1). In the first step, the image processing module is trained using instances of seen classes. After training, the classifier weights for seen classes have semantic knowledge. However, the weights for unseen classes are random. We want to align the seen class weights with their GPM weight counterpart. The GPM weights contain semantic knowledge about the seen classes as well as the unseen classes. The unseen class weight of GPM are semantically richer compared to the random weights from our image processing module.

In the second step, we train the *GPM*. The supervision for training the *GPM* comes from weights of seen labels from step-1. The seen class weights have a high semantic knowledge obtained by processing the images. These can be used to enrich the features for corresponding labels in the *GPM*. Please note that supervision is provided for only the seen labels but owing to the nature of graph convolution layers (Kipf and Welling, 2016), the unseen class features are simultaneously enriched. Once converged, the enriched *GPM* weights replace the classifier weights in the image processing module. This weight replacement ensures a transfer of rich semantic knowledge for the unseen classes. Finally, we fix the classifier weights (which are the GPM weights we copied), and fine-tune the image processing module layers. This ensures that the classifier weights are semantically meaningful to the visual features from the vision backbone.

Image processing module. We use *DenseNet121* (Huang et al., 2018) with a fully-connected layer with 1024-dimension. The weights learned by this fully connected layer $W_\phi \in R^{1024\times C}$, are considered to be the image representation of a radiograph. Since the

	k=2			k=3			AUROC		
Method	**p@k**	**r@k**	**f1@k**	**p@k**	**r@k**	**f1@k**	**S**	**U**	**HM**
NIH Chest X-ray									
CNN	0.28	0.34	0.30	0.23	0.43	0.29	**0.80**	0.52	0.63
CXR-ML-GZSL	0.33	**0.36**	0.32	0.28	**0.47**	0.34	**0.79**	0.66	0.72
Ours	**0.38**	0.33	**0.35**	**0.31**	0.43	**0.36**	0.79 $\pm$0.001	**0.68** $\pm$ 0.002	**0.73** $\pm$ 0.001
Indiana University Chest X-ray									
CNN	0.23	0.25	0.24	0.27	0.34	0.30	**0.70**	0.68	0.69
CXR-ML-GZSL	**0.33**	0.26	**0.29**	0.27	0.35	0.31	0.68	0.79	0.73
Ours	0.28	**0.28**	0.28	**0.28**	**0.36**	**0.32**	0.68 $\pm$0.001	**0.80** $\pm$0.002	0.74 $\pm$0.001

Table 1: Performance Evaluation on the NIH Chest X-ray and Indiana University Chest X-ray dataset. We report the results using Precision@k, Recall@k , F1@k for k $\in \{2, 3\}$. We also report AUROC for *seen (S) & unseen (U) classes and the Harmonic Mean (HM)*. CXR-ML-GZSL refers to (Hayat et al., 2021) and *CNN* is DenseNet121 trained on only the seen classes. We report the mean and standard deviation value across five runs of the model. Please refer to the appendix for more details.

model sees instances of only the seen classes, the representation is meaningful only for them (seen classes). The weights are *random* for the unseen classes. Thus, developing the capacity to handle unseen diseases can then be expressed as predicting a *new* set of weights for each of the unseen classes.

While training the image processing module, a weighted multi-label classification loss $\mathcal{L}_{cls}$(Eq. 1) is used to account for the potential data-imbalance (Chen et al., 2020a). The weights are adjusted to account for a surplus of positive or negative samples in a mini-batch.

$$\mathcal{L}_{cls} = -\omega_p \sum_{l_i=1} log(\sigma(\hat{y}_i)) - \omega_n \sum_{l_i=0} log(1 - \sigma(\hat{y}_i)) \tag{1}$$

$\hat{y}_i$ is the model logit, l_i is the corresponding label, $|P|$ and $|N|$ are the total number of positive and negative samples per mini-batch. Thus, $\omega_p = \frac{|P|+|N|+1}{|P|+1}$ and $\omega_n = \frac{|P|+|N|+1}{|N|+1}$ are the balancing factors to handle data imbalance.

Graph construction. We use the Unified Medical Language System (*UMLS*) (Bodenreider, 2004) as the KG of our choice to obtain semantic clinical information. Since, the KG is huge, containing millions of entries, often not directly related to the task at hand, a naive parsing of the entire KG is neither feasible nor beneficial. Thus, we parse *only* for a subset of relations based on prior medical knowledge. These relations include: inverse_isa, finding_site_of, part_of, is_associated_anatomic_site_of and has_member.

Note that we none of the relationships explicitly encode the co-occurrences between these diseases. Explicitly introducing such co-occurrences as inductive bias will lead to data leakage, affecting the generalized zero shot learning paradigm. However, it may be beneficial for the model to recognize that some diseases often occur together. However, if such relationships are learnt, it is completely data driven and utilizes the semantic knowledge from the graph as well as the image information.

Figure 2 summarizes the three steps to parse this subgraph. First, we extract the entities (nodes) corresponding to the set of target classes (i.e. both seen and unseen diseases).

Starting from each of these entities, we extract its 5-hop neighbourhood, resulting in a first noisy subgraph. (Please refer to the appendix for details about parsing a k-hop neighbourhood). This subgraph is then trimmed using *all-pair-shortest path* between the seen and unseen labels. After the trimming operation, seen nodes, unseen nodes, and nodes on the shortest path between them remain. All these nodes are initialized with BioBERT embeddings creating the graph G (Alsentzer et al., 2019).

Graph processing module (GPM). The *GPM* aims to enrich features of the parsed subgraph G. The *GPM* is realized using a series of graph convolution layers (Brody et al., 2021). Assume w_G^j denotes the BioBert representation of the j^{th} node in the parsed graph. This representation is enriched using a series of GATv2 layers (Brody et al., 2021) in the following layout:

$$w_G^j \rightarrow GATv2 \rightarrow ReLU \rightarrow GATv2 \rightarrow ReLU \rightarrow GATv2 \rightarrow \hat{w}_G^j \in R^{1024}$$

We concatenate the enriched representations corresponding to the C classes in our dataset, resulting in $W_G \in R^{1024 \times C}$. These disease representations based on the graph are referred to as the GPM weights. Next, we align the weights of the seen classes between the *GPM* and the image processing module, i.e.

$$\mathcal{L}_{reg} = \sum_{j \in seen} ||W_\phi^j - W_G^j||^2 \tag{2}$$

where W_ϕ^j is the corresponding representation of the j^{th} disease from the image processing module. The loss is computed *only* for the seen classes.

The final step involves copying over the GPM weights W_G to the image processing module. With the updated weights, the image processing module has knowledge about both seen and unseen classes. Please note that the *GPM* module is not used during inference. Only the image processing module is required to classify the unseen test samples.

4. Experiment

Dataset We evaluate our method on two public chest X-ray datasets: a) The NIH Chest X-ray dataset (Wang et al., 2017), and b) The Indiana Univ Chest X-ray dataset (Shin et al., 2016). Radiographs with multi-label annotations are provided for both datasets.

NIH Chest X-ray. 112,120 frontal X-ray images are split into training (70%), validation (10%) and test sets (20%). Each image is associated with 14 class labels. We use *Atelectasis, Effusion, Infiltration, Mass, Nodule, Pneumothorax, Consolidation, Cardiomegaly, Pleural Thickening, and Hernia* as the *seen* classes while *Edema, Pneumonia, Emphysema, and Fibrosis* are the *unseen* classes, resulting in 30,758 training images, 4,474 validation images and 10,510 test images, same as (Hayat et al., 2021).

Indiana University Chest X-ray. We used a similar setup as the NIH dataset. We split the frontal X-ray images into training (70%), validation (10%) and test sets (20%). Each image is associated with 17 class labels. We use *Cardiomegaly, Scoliosis, Effusion, Thickening, Pneumothorax,Hernia, Calcinosis, Atelectasis, Cicatrix, Opacity, Lesion,Airspace disease, and Hypoinflation* as the *seen* classes while *Edema, Pneumonia, Emphysema, Fibrosis* are

the *unseen* classes, resulting in 1014, 145, and 408 for training, validation, and test sets respectively.

Evaluation metrics. We report overall precision, recall, and f1 scores for the top k predictions (where $k \in 2, 3$) and the average area under the receiving operating characteristic curve (AUROC) for *seen* and *unseen* classes and their harmonic mean.

4.1. Comparison with state-of-the-art

We summarize the results in comparison with existing methods in Table 1. Our model performs better than the baseline for *unseen* classes while performing comparably on the seen classes. Since our proposed solution relies on a universal knowledge graph (*UMLS*) and is not tightly coupled to the dataset we operate on, the extension of our method to different datasets with different numbers of target labels is almost trivial. Verifying this, we evaluate the baseline and our proposed method on the Indiana University Chest X-ray dataset. Note that another UMLS sub-graph has to be created as the label set changes. The remaining modules, however, remain unchanged. Observe the improvement over baseline performance, showcasing our methodś extensibility with minimal changes.

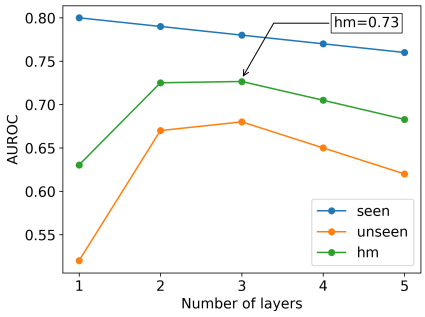

Figure 3: Plots of AUROC values vs. the number of GATv2 layers in the Graph Processing Module (GPM). The Harmonic Mean (HM) of AUROC for *seen* & *unseen* classes tends to increase first reaching a maximum value of 0.73 for three GATv2 layers and then decreases.

	AUROC		
Method	**S**	**U**	**HM**
CNN	**0.80**	0.52	0.63
BERT	0.78	0.60	0.68
Random Graph -ER + BERT	0.77	0.58	0.66
Random Graph -SBM + BERT	0.78	0.57	0.66
Random Graph -PAM + BERT	0.77	0.51	0.61
UMLS + Word2Vec	0.78	0.61	0.69
UMLS + BERT	0.79	**0.68**	**0.73**

Table 2: Ablation study. The **CNN** model is trained only based on the *seen* classes. **BERT** model used *BioBERT* embeddings for the nodes but assumes no graph structure. **Random Graph + BERT + *** uses a graph created from random graph generation algorithms and uses *BioBERT* embeddings for its nodes. **UMLS + Word2Vec** uses G_{UMLS} but initializes the node embeddings using Bio-Word2Vec. **UMLS + BERT** uses the G_{UMLS} (UMLS parsing + BioBERT node embeddings).

4.2. Ablation Study

To hightlight our contributions and evaluate different components, we run the ablation experiment on the NIH Chest X-ray dataset. Please refer to supplementary for detail discussions about the *CNN* baseline method.

BioBERT embeddings vs. knowledge graph. It is known that *BioBERT* embeddings are semantically rich in text representation. However, they might not sufficiently capture clinical relation information in the GZSL setting. We ran an experiment using the BioBERT embeddings but without the graph structure. The nodes are initialized with BioBERT embeddings and passed through several fully connected layers, processing nodes independently without any inter-node interaction. The semantic richness ensures decent performance on the *unseen* classes (AUROC 0.60), obtaining a HM of 0.68 overall. However, the performance is still considerably worse than our proposed graph for *unseen* classes (**0.60** vs. **0.68**), indicating that the BioBERT embeddings are insufficient to bridge the semantic gap.

Learned graph vs. random graph. To analyze the importance of graph structure, we replace the *UMLS* graph with different random graphs (Stochastic Block Model, Planted Partition Model and Erdos Renyi random graph model (Newman et al., 2002)). As can be seen in Table 2, all random graph models perform worse than the BioBERT embedding model. We attribute this to an incoherent graph structure in random graphs, leading to a *negative knowledge transfer* between the nodes. The decrease in performance is especially steep in the case of *unseen* classes. This is expected since the learned graph structure passes essential semantic knowledge to classify unseen diseases and it indicates that the graph structure is critical for the overall performance.

The importance of node embeddings. To evaluate the importance of node embeddings, we initialize the G_{UMLS} nodes using BioWord2Vec embeddings (Zhang et al., 2019), instead of BioBERT embeddings. On average, the model performs better than the independently processed BioBERT embeddings. Still, the performance is much worse compared to the proposed solution (**0.68** vs. **0.61** for *unseen* classes). These experiments corroborate the importance of graph structure and strong feature representation for the node embeddings. Hence, the proposed solution uses UMLS graph structure and BioBERT embeddings.

The effect of the depth of GAT layers. The GPM module uses GATv2 convolutions to process node embeddings. We experimented with a different number of convolution layers, and results are shown in Figure 3. As we can observe, the AUROC value is maximum when using *three* GATv2 layers with an HM of 0.73. From Table 4 in the supplementary, we can see that the maximum distance between any two target nodes is four. Hence, with 3 layers, neighbourhood aggregation covers the entire graph and additional layers lead to performance degradation possibly due to the over-smoothing effect (Chen et al., 2020b).

4.3. Model interpretability

We use the *GNNExplainer* (Ying et al., 2019) to get more insights about predictions made by the model. It would produce a subgraph G_S by pruning some of the nodes of the original graph. Nodes important for downstream task are retained while extraneous nodes are pruned away. Figure 4 shows the result for the node *lung mass*. We observe that graph connections are not discarded completely. This shows that individual node features, by themselves, are insufficient for the downstream prediction. Furthermore, while predicting *lung mass*, more importance is given to nodes representing lung morphology and lung disease

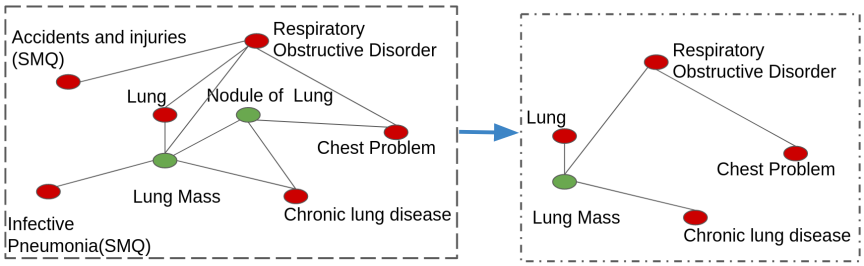

Figure 4: Output from the GNNExplainer. Nodes colored in green are the target seen and unseen labels for the NIH Chest X-ray dataset, while the nodes in red represent the extra labels obtained by parsing the UMLS. The graph structure is not discarded completely after pruning. This shows that the individual node features, by themselves, are insufficient for the downstream task.

while nodes such as Accidents and Injuries (SMQ) are pruned away. (Please refer to the appendix for more details as well as GradCam (Selvaraju et al., 2017) visualization)

5. Conclusion

We propose a novel solution for parsing, storing and processing medical knowledge graphs (e.g. *UMLS*) to improve generalized zero shot learning. We also show that our method can be easily extended to multiple datasets with minimal effort. We find that knowledge graphs provide a very rich source of semantic information that can be used for diseases not seen during training. A limitation of this work is that we have only used the *structural* information from the *KG* and considered it as a homogeneous graph. As such, we do not differentiate if two medical concepts are related in distinct ways (e.g. *finding-site-of vs. part of* etc). In future work, we aim to treat the KG as heterogeneous (i.e., treating different relations independently), thereby further enriching the semantic knowledge transfer as well as to check for different combinations of the seen and unseen disease pairs.

References

Emily Alsentzer, John Murphy, William Boag, Wei-Hung Weng, Di Jin, Tristan Naumann, and Matthew McDermott. Publicly available clinical BERT embeddings. In *Proceedings of the 2nd Clinical Natural Language Processing Workshop*, pages 72–78, Minneapolis, Minnesota, USA, June 2019. Association for Computational Linguistics. doi: 10.18653/v1/W19-1909. URL https://www.aclweb.org/anthology/W19-1909.

Albert-László Barabási and Réka Albert. Emergence of scaling in random networks. *science*, 286(5439):509–512, 1999.

Olivier Bodenreider. The unified medical language system (umls): integrating biomedical terminology. *Nucleic acids research*, 32(suppl_1):D267–D270, 2004.

Shaked Brody, Uri Alon, and Eran Yahav. How attentive are graph attention networks?, 2021.

Bingzhi Chen, Jinxing Li, Guangming Lu, Hongbing Yu, and David Zhang. Label co-occurrence learning with graph convolutional networks for multi-label chest x-ray image classification. *IEEE journal of biomedical and health informatics*, 24(8):2292–2302, 2020a.

Deli Chen, Yankai Lin, Wei Li, Peng Li, Jie Zhou, and Xu Sun. Measuring and relieving the over-smoothing problem for graph neural networks from the topological view. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 3438–3445, 2020b.

Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein. *Introduction to Algorithms, 3rd Edition*. MIT Press, 2009. ISBN 978-0-262-03384-8. URL http://mitpress.mit.edu/books/introduction-algorithms.

Jeffrey De Fauw, Joseph R Ledsam, Bernardino Romera-Paredes, Stanislav Nikolov, Nenad Tomasev, Sam Blackwell, Harry Askham, Xavier Glorot, Brendan O'Donoghue, Daniel Visentin, et al. Clinically applicable deep learning for diagnosis and referral in retinal disease. *Nature medicine*, 24(9):1342–1350, 2018.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

Andre Esteva, Brett Kuprel, Roberto A Novoa, Justin Ko, Susan M Swetter, Helen M Blau, and Sebastian Thrun. Dermatologist-level classification of skin cancer with deep neural networks. *nature*, 542(7639):115–118, 2017.

Yoav Goldberg and Omer Levy. word2vec explained: deriving mikolov et al.'s negative-sampling word-embedding method. *arXiv preprint arXiv:1402.3722*, 2014.

Nasir Hayat, Hazem Lashen, and Farah E. Shamout. Multi-label generalized zero shot learning for the classification of disease in chest radiographs, 2021.

Gao Huang, Zhuang Liu, Laurens van der Maaten, and Kilian Q. Weinberger. Densely connected convolutional networks, 2018.

Thomas N Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*, 2016.

Aoxue Li, Zhiwu Lu, Jiechao Guan, Tao Xiang, Liwei Wang, and Ji-Rong Wen. Transferrable feature and projection learning with class hierarchy for zero-shot learning. *International Journal of Computer Vision*, 128:2810–2827, 2020.

Mark EJ Newman, Duncan J Watts, and Steven H Strogatz. Random graph models of social networks. *Proceedings of the national academy of sciences*, 99(suppl 1):2566–2572, 2002.

Angshuman Paul, Thomas C Shen, Sungwon Lee, Niranjan Balachandar, Yifan Peng, Zhiyong Lu, and Ronald M Summers. Generalized zero-shot chest x-ray diagnosis through trait-guided multi-view semantic embedding with self-training. *IEEE Transactions on Medical Imaging*, 2021.

Pranav Rajpurkar, Jeremy Irvin, Kaylie Zhu, Brandon Yang, Hershel Mehta, Tony Duan, Daisy Ding, Aarti Bagul, Curtis Langlotz, Katie Shpanskaya, et al. Chexnet: Radiologist-level pneumonia detection on chest x-rays with deep learning. *arXiv preprint arXiv:1711.05225*, 2017.

Timo Schick and Hinrich Schütze. Rare words: A major problem for contextualized embeddings and how to fix it by attentive mimicking. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 8766–8774, 2020.

Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE international conference on computer vision*, pages 618–626, 2017.

Hoo-Chang Shin, Kirk Roberts, Le Lu, Dina Demner-Fushman, Jianhua Yao, and Ronald M Summers. Learning to read chest x-rays: Recurrent neural cascade model for automated image annotation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2497–2506, 2016.

Xiaolong Wang, Yufei Ye, and Abhinav Gupta. Zero-shot recognition via semantic embeddings and knowledge graphs. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

Xiaosong Wang, Yifan Peng, Le Lu, Zhiyong Lu, Mohammadhadi Bagheri, and Ronald M Summers. Chestx-ray8: Hospital-scale chest x-ray database and benchmarks on weakly-supervised classification and localization of common thorax diseases. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2097–2106, 2017.

Yongqin Xian, Bernt Schiele, and Zeynep Akata. Zero-shot learning-the good, the bad and the ugly. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4582–4591, 2017.

Rex Ying, Dylan Bourgeois, Jiaxuan You, Marinka Zitnik, and Jure Leskovec. Gnn explainer: A tool for post-hoc explanation of graph neural networks. *arXiv preprint arXiv:1903.03894*, 2019.

Yijia Zhang, Qingyu Chen, Zhihao Yang, Hongfei Lin, and Zhiyong Lu. Biowordvec, improving biomedical word embeddings with subword information and mesh. *Scientific data*, 6(1):1–9, 2019.

Bo Zhao, Xinwei Sun, Yuan Yao, and Yizhou Wang. Zero-shot learning via shared-reconstruction-graph pursuit. *arXiv preprint arXiv:1711.07302*, 2017.

Appendix A. Model Interpretability

Grad-CAM Figure 5 shows some of the visualizations obtained using Grad-CAM on samples containing *unseen* classes in the test set. As we can see, our model focuses on radiograph regions most likely responsible for the diseases.

Disease	**Nearest Neighbour (*BioBERT*)**	**Nearest Neighbour (GPM)**
Atelectasis	Lung Problem	Pneumonia
Cardiomegaly	Chest problem	Diaphragmatic Hernia
Pleural effusion	Pleural Diseases	Thickening of pleura
Pulmonary Infiltrate	Lower respiratory tract structure	Pneumonia
Lung mass	Lung diseases	Abnormal pleura morphology
Nodule of lung	Lesion of lung	Thickening of pleura
Pneumonia	Lung Problem	Pulmonary Edema
Pneumothorax	Pulmonary Emphysema	Pulmonary Emphysema
Lung consolidation	Lung diseases	Interstitial lung disease (SMQ)
Pulmonary Edema	Lung Problem	Pneumonia
Pulmonary Emphysema	Pulmonary Fibrosis	Diaphragmatic Hernia
Pulmonary Fibrosis	Pulmonary Emphysema	Diaphragmatic Hernia
Thickening of pleura	Disorder of pleura and pleural cavity	Pulmonary Edema
Diaphragmatic Hernia	Respiratory Diaphragm	Pulmonary Emphysema

Table 3: Comparing the 1-nearest neighbours in the embedding space for *BioBERT* vs. *GPM* feature space embeddings. While *BioBERT*'s embedding space is valid but generic, the *GPM* feature space is aligned to learn the relationship between different diseases based on the *UMLS* structure.

Feature Space Lookup Nearest Neighbour lookup in the feature space is an efficient way to decipher the predictions made by a Deep Learning model. In Table 3 we explore the feature space of original *BioBERT* embeddings and the embeddings produced by *GPM*. We use an L2 distance-based 1-Nearest Neighbour (*NN*) lookup. The *BioBERT* feature space has a lot of semantic information, but it does inherently know the relationship between different diseases. For instance, in its embedding space, *NN* of *Pleural Effusion* is *Pleural disease*. Although this is valid but the information does not include relations between these diseases. The *GPM*, on the other hand, brings *Thickening of Pleura* closer to *Pleural Effusion* in the embedding space, thereby explicitly learning a relationship between the two. This demonstrates that a feature space with rich semantic features and efficacious encoding between diseases is learned by our model.

GNNExplainer Since a graph provides inherent explainability, we determine what nodes and edges in the graph are considered relevant for predictions using the *GNNExplainer* (Ying et al., 2019) framework. The GNNExplainer would produce a subgraph G_S by pruning some of the nodes of the original graph. X_S^F are the node features of the resulting subgraph. We compute the mean square error between the original GPM node features x^{jd} and the resulting subgraph node features $\tilde{x_S}^{jd}$ (referred to as the *feature_regression_loss*). We define $H(Y|G = G_S, X = X_S^F)$ as the entropy of the subgraph. It encodes how much information is "lost" by removing the nodes (& their associated features) from the original graph. We aim to find such nodes that can be removed with minimal change in the expressivity of the model. Conversely, these nodes play a minimal role in the model decision and hence, for understanding the model behavior, we should not focus on them (Ying et al., 2019). Remov-

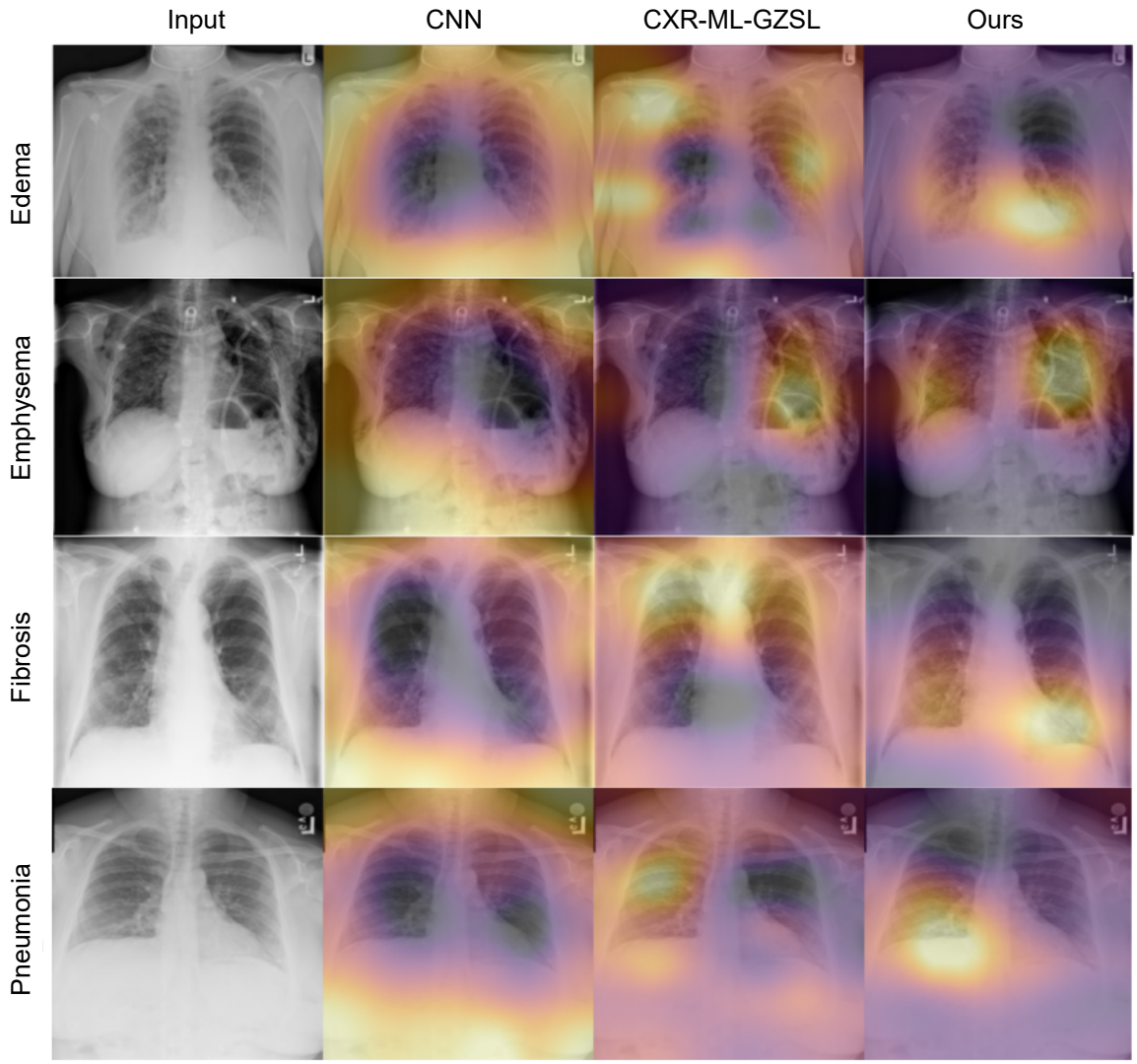

Figure 5: Saliency map visualization for the *unseen* classes. Each row contains one of the unseen diseases and the Grad-CAM output of the three models. We have included the original input image in the first column for reference.The model focuses on regions that are relevant for diagnosis of the individual diseases.

ing such nodes would lead to minimal changes to the entropy & the *feature_regression_loss*. Thus, to select only the consequential nodes in the graph, we optimize

$$\mathcal{L}_{exp} = \lambda \cdot \frac{1}{D} \sum_{j} \sum_{D} (x^{jd} - \tilde{x_S}^{jd})^2 + H(Y|G = G_S, X = X_S^F) \tag{3}$$

We empirically set λ to 10^3 to ensure that all loss terms are approximately of the same scale. Figure 4 visualizes some nodes in the *UMLS* graph. We observe that graph

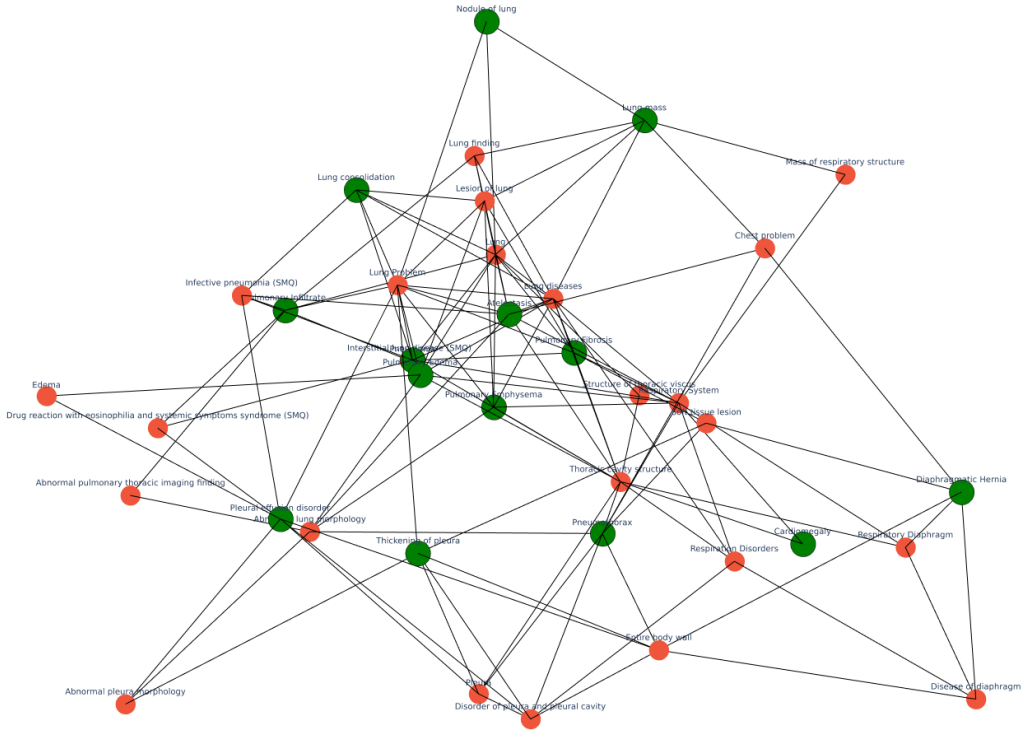

Figure 6: (Best viewed in zoom) A visualization of the parsed graph. The nodes colored in green are the target labels for the NIH Chest X-ray dataset, while those colored in red are the extra labels obtained by parsing the *UMLS*.

connections are not discarded completely. This shows that individual node features, by themselves, are insufficient for the downstream prediction. Furthermore, while predicting *lung mass*, more importance is given to nodes representing lung morphology and lung disease while nodes such as Accidents and Injuries (SMQ) are pruned away.

A.1. Shortest Distance between the nodes

Table 4 summarizes the pair-wise distance between all the target labels for the NIH Chest X-ray dataset. As we can see, there are **no self-loops** in the graph and the maximum distance between two target labels is **4**. Hence, the *GPM* should produce the best result for 3 conv layers. We observe the same in Figure 3.

Appendix B. Implementation Details

The training of our model happens in three steps. In the first step, the vision backbone is trained for 40 epochs. Next we train the GPM for for 1000 iterations. Finally, we finetune the model on each dataset in a supervised manner. The experiments are conducted using the PyTorch Geometric on a NVIDIA GeForce RTX 3090 machine.

Node	**Node**													
	A	C	PED	PI	LM	N	Pn	LC	Pt	PEdm	PEpy	PF	T	D
A	0	2	2	2	2	2	2	2	2	2	2	2	2	2
C	2	0	3	**4**	3	**4**	2	3	2	2	2	2	3	3
PED	2	3	0	2	2	2	2	2	2	2	2	2	2	2
PI	2	**4**	2	0	2	2	2	2	3	2	2	2	2	**4**
LM	2	3	3	2	0	1	2	2	2	2	2	2	3	2
N	2	**4**	2	2	1	0	2	2	3	2	2	2	2	3
Pn	2	2	2	2	2	2	0	1	2	2	2	2	2	3
LC	2	3	2	2	2	2	1	0	3	2	2	2	2	**4**
Pt	2	2	2	3	2	3	2	3	0	2	2	2	2	2
PEdm	2	2	2	2	2	2	2	2	2	0	2	2	2	3
PEpy	2	2	2	2	2	2	2	2	2	2	0	2	2	3
PF	2	2	2	2	2	2	2	2	2	2	2	0	2	2
T	2	3	2	2	3	2	2	2	2	2	2	2	0	2
D	2	3	2	**4**	2	3	3	**4**	2	3	3	2	2	0

Table 4: All pair shortest path between the target label nodes for NIH Chest X-ray dataset. **A** represents Atelectasis, **C** represents Cardiomegaly, **PED** represents Pleural Effusion Disorder, **PI** represents Pulmonary Infiltrate, **LM** represents Lung Mass, **N** represents Nodule of lung, **Pn** represents Pneumonia, **LC** represents Lung Consolidation, **Pt** represents Pneumothorax, **PEdm** represents Pulmonary Edema, **PEpy** represents Pulmonary Emphysema, **PF** represents Pulmonary Fibrosis, **T** represents Thickening of pleura, **D** represents Diaphragmatic Hernia. As we can see, the maximum distance between the target label nodes is **4** and thus, using 4 convolution layers would lead to an oversmoothing effect for the *label-set nodes*.

Random Graph Generation. In the table we report results for the Erdos-Renyi model generated using edge-probability value of 0.2. For the Stochastic Block model, we use a block-size of $\frac{1}{num\_classes}$ and an edge-probability of 0.2. (While we report results for probability value of 0.2, the experiments were repeated for values ranging from 0.1 to 0.5. Best results were obtained when we set the value to 0.2 though). For the Planted Partition Model we use the Barabasi- Albert-Graph generation (Barabási and Albert, 1999). In all the cases, the number of nodes is the same as G. We repeated the graph generation process 10 times with different seed values.

Appendix C. Results

Class-wise AUROC comparison Table 5 shows the per-class AUROC value for the test-set. As we can see, our method tends to perform better for the *unseen* classes and is quite close to the baseline for the samples from *seen* classes.

Method	Atelectasis	Cardiomegaly	Pleural Effusion Disorder	Pulmonary Infiltrate	Lung Mass	Nodule of lung	Pneumothorax	Lung Consolidation	Thickening of pleura	Diaphragmatic Hernia	**Pneumonia**	**Pulmonary Edema**	**Pulmonary Emphysema**	**Pulmonary Fibrosis**
CNN	0.77	**0.91**	0.83	**0.71**	0.80	0.77	0.84	0.72	**0.74**	**0.96**	0.51	0.51	0.45	**0.60**
CXR-ML-ZSL	0.76	0.90	0.83	0.70	0.80	0.75	0.83	0.69	0.72	0.90	0.62	0.67	0.74	**0.60**
Ours	**0.79**	0.90	0.83	**0.71**	**0.82**	**0.79**	**0.85**	**0.73**	0.67	0.81	**0.66**	**0.70**	**0.80**	0.58

Table 5: The Class-wise AUROC comparison across all disease classes in the test set. As we can see, our method tends to obtain the best results for the *unseen* classes (marked in bold) while being comparable to the *seen* classes.

Appendix D. Graph Construction Process

The knowledge graph is constructed by parsing the Unified Medical Language System (UMLS) (Bodenreider, 2004). We parse the *MRREL.RRF* file in order to obtain the different relations between entities. The empty entries are filtered. We also remove entries of the relation type DEL, XR, RL which denote deleted relations, no-mapping relations and self-relations. All remaining entries are viable candidate relations. Let us denote these filtered relationships as $R_{filtered}$

In the next step, we start with the seed labels. These seed labels are the target diseases, specific to the dataset. We look up these labels in the $\mathrm{R_{filtered}}$. We limit the lookup to only five relation-types, namely inverse_is_a, finding_site_of, part_of, has_member and is_associated_anatomic_site_of.

The entities obtained from the lookup forms the one-hop neighbourhood of our seed labels. We use the idea of depth first search in order to obtain the k-hop neighbourhood. We implement depth first search using its recursive formulation (Cormen et al., 2009). Specifically, we keep track of the concepts visited in the current lookup and use them as seeds for the next round of lookup. We repeat the process k times.

Finally, we filter away duplicate relationships. We also remove relationships that contain descriptions in language other than English. Noisy entries such as those containing only white spaces are removed using a python based regex matcher. Finally, all the remaining entries form our graph structure.

We initialize the concepts with their BioBERT embeddings. While using BioWord2Vec model in our ablations, we used UNK token embeddings for the words that did not have a vocabulary mapping. This forms our graph G. However, the graph is huge and difficult to process. Hence, we use the all-pair shortest path formulation to further trim the graph.