Learning and inferencing challenges in human-in-the-loop decision systems

by

Deepesh Agarwal

B. Tech., Institute of Technology, Nirma University, 2018

M. Tech., Indian Institute of Technology (IIT) Gandhinagar, 2020

AN ABSTRACT OF A DISSERTATION

submitted in partial fulfillment of the
requirements for the degree

DOCTOR OF PHILOSOPHY

Mike Wiegers Department of Electrical and Computer Engineering
Carl R. Ice College of Engineering

KANSAS STATE UNIVERSITY
Manhattan, Kansas

2024

Abstract

The computational capabilities of AI engines integrated with human knowledge and experience can help create intelligent human-in-the-loop (HITL) decision systems. In safety-critical applications that require a certain level of human supervision, human and AI engine errors can be costly. Thus, it is crucial to identify challenges prevailing at several levels of HITL decision systems that hinder the learning and inferencing processes, and subsequently address them within the learning scheme. This dissertation designates the learning and inferencing challenges in HITL systems at three different levels, namely, representation-level, feature-level and model-level challenges and addresses these challenges within an Active Learning (AL) context.

There are several hindrances, such as unavailability of labels for the AL algorithm at the beginning; unreliable external source of labels during the querying process; or incompatible mechanisms to evaluate the performance of Active Learner. Inspired by these practical challenges, this dissertation presents a hybrid query strategy-based AL framework that addresses three practical challenges simultaneously: cold-start, oracle uncertainty and performance evaluation of Active Learner in the absence of ground truth. The heuristics obtained during the querying process serve as the fundamental premise for accessing the performance of Active Learner. The idea of AL is further extended to representation learning in non-Euclidean space like graphs as well. Both node attributes and topological information are incorporated in the learning scheme. The node features are exploited while training the GNN-based decision model and topological information is considered during selective sampling of the nodes.

Modeling human behavior in collaborative human-AI decision setup is not straightforward. This dissertation, for the first time, presents a systematic framework for simulation, modeling, tracking and adaptation of behavioral biases in a collaborative HITL decision

environment within an AL context. The issue of poor generalization performance and overfitting of decision models is addressed by incorporating observational biases while training the decision models. This dissertation presents two case studies demonstrating ways to incorporate observational biases within the learning frameworks. Despite the fact that AI-powered systems have provided competitive benefits in the recent years, the black-box nature prohibits the explainability of their decisions and drives them to lack transparency. This issue prompted the development of explainable artificial intelligence (XAI), which supports AI systems that can explain their internal processes and decision-making methods. This dissertation presents two case studies to demonstrate different ways of explaining the predictions made by decision models.

Conventional NN do not furnish uncertainty estimates associated with their predictions, and are therefore ill-calibrated. Uncertainty quantification techniques offer probability distributions or confidence intervals to represent the uncertainty associated with NN predictions, instead of solely presenting the point predictions/estimates. Once the uncertainty in NN is quantified, it is crucial to leverage this information to modify training objectives and improve accuracy and reliability of the corresponding decision models. This dissertation establishes a novel framework to utilize the knowledge of input and output uncertainties in NN to guide querying process in the context of Active Learning. The lower and upper bounds for label complexity are derived analytically.

The methods proposed in this dissertation are highly beneficial for safety-critical applications, that demand significant human monitoring and any error due to human and AI components can be expensive. For example, an effective and rigorous decision support tool in medical diagnosis can help doctors/clinicians nudge the possibility of prescribing further tests for better diagnosis, thereby making a well-informed decision with higher confidence.

Learning and inferencing challenges in human-in-the-loop decision systems

by

Deepesh Agarwal

B. Tech., Institute of Technology, Nirma University, 2018

M. Tech., Indian Institute of Technology (IIT) Gandhinagar, 2020

A DISSERTATION

submitted in partial fulfillment of the
requirements for the degree

DOCTOR OF PHILOSOPHY

Mike Wiegers Department of Electrical and Computer Engineering
Carl R. Ice College of Engineering

KANSAS STATE UNIVERSITY
Manhattan, Kansas

2024

Approved by:

Major Professor
Dr. Balasubramaniam Natarajan

Copyright

Abstract

The computational capabilities of AI engines integrated with human knowledge and experience can help create intelligent human-in-the-loop (HITL) decision systems. In safety-critical applications that require a certain level of human supervision, human and AI engine errors can be costly. Thus, it is crucial to identify challenges prevailing at several levels of HITL decision systems that hinder the learning and inferencing processes, and subsequently address them within the learning scheme. This dissertation designates the learning and inferencing challenges in HITL systems at three different levels, namely, representation-level, feature-level and model-level challenges and addresses these challenges within an Active Learning (AL) context.

There are several hindrances, such as unavailability of labels for the AL algorithm at the beginning; unreliable external source of labels during the querying process; or incompatible mechanisms to evaluate the performance of Active Learner. Inspired by these practical challenges, this dissertation presents a hybrid query strategy-based AL framework that addresses three practical challenges simultaneously: cold-start, oracle uncertainty and performance evaluation of Active Learner in the absence of ground truth. The heuristics obtained during the querying process serve as the fundamental premise for accessing the performance of Active Learner. The idea of AL is further extended to representation learning in non-Euclidean space like graphs as well. Both node attributes and topological information are incorporated in the learning scheme. The node features are exploited while training the GNN-based decision model and topological information is considered during selective sampling of the nodes.

Modeling human behavior in collaborative human-AI decision setup is not straightforward. This dissertation, for the first time, presents a systematic framework for simulation, modeling, tracking and adaptation of behavioral biases in a collaborative HITL decision

environment within an AL context. The issue of poor generalization performance and overfitting of decision models is addressed by incorporating observational biases while training the decision models. This dissertation presents two case studies demonstrating ways to incorporate observational biases within the learning frameworks. Despite the fact that AI-powered systems have provided competitive benefits in the recent years, the black-box nature prohibits the explainability of their decisions and drives them to lack transparency. This issue prompted the development of explainable artificial intelligence (XAI), which supports AI systems that can explain their internal processes and decision-making methods. This dissertation presents two case studies to demonstrate different ways of explaining the predictions made by decision models.

Conventional NN do not furnish uncertainty estimates associated with their predictions, and are therefore ill-calibrated. Uncertainty quantification techniques offer probability distributions or confidence intervals to represent the uncertainty associated with NN predictions, instead of solely presenting the point predictions/estimates. Once the uncertainty in NN is quantified, it is crucial to leverage this information to modify training objectives and improve accuracy and reliability of the corresponding decision models. This dissertation establishes a novel framework to utilize the knowledge of input and output uncertainties in NN to guide querying process in the context of Active Learning. The lower and upper bounds for label complexity are derived analytically.

The methods proposed in this dissertation are highly beneficial for safety-critical applications, that demand significant human monitoring and any error due to human and AI components can be expensive. For example, an effective and rigorous decision support tool in medical diagnosis can help doctors/clinicians nudge the possibility of prescribing further tests for better diagnosis, thereby making a well-informed decision with higher confidence.

Table of Contents

List of Figures

List of Tables

Nomenclature

AI	Artificial Intelligence
AL	Active Learning
ML	Machine Learning
DL	Deep Learning
GNN	Graph Neural Network
EGR	Effective Graph Resistance
DNN	Deep Neural Network
DSMRS	Dissimilarity-based Sparse Modeling Representative Selection
BELGAM	Bayesian Deep Latent Gaussian Model
US	Uncertainty Sampling
QBC	Query-by-Committee
DWM	Density-weighted Methods
EC	Entropy of the Class probabilities
MU	Margin Uncertainty
CU	Classifier Uncertainty
CE	Consensus Entropy of the committee of classifiers
IE	Information density with Euclidean distance
IC	Information density with Cosine similarity
HMI	Human Machine Interface
CSTR	Continuous Stirred Tank Reactor
WPD	Wavelet Packet Decomposition
WCSS	Within-Cluster Sum of Squares distance
kNN	k-Nearest Neighbors
DGL	Deep Graph Library

MLP	Multi-Layer Perceptron
CC	Clustering Coefficient
CPHS	Cyber-Physical-Human Systems
ESIS	Exploitation-Scrutinization-Investigation-Scrutinization
HITL	Human-In-The-Loop
GAN	Generative Adversarial Networks
XAI	Explainable Artificial Intelligence
SSL	Self Supervised Learning
TBI	Traumatic Brain Injury
BBB	Blood-Brain Barrier
CT	Computed Tomography
MRI	Magnetic Resonance Imaging
EV	Extracellular Vesicles
LFQ	Label-free quantification
PPI	Protein-Protein Interaction
hpi	hours post injury
dpi	days post injury
MSA	Multi-head Self-Attention
LN	Layer Normalization
FNN	Feed-forward Neural Network
RC	Residual Connections
GeLU	Gaussian error Linear Unit)
NLP	Natural Language Processing
BERT	Bidirectional Encoder Representations from Transformers
SVM	Support Vector Machine
RF	Random Forest
CNN	Convolutional Neural Networks

RNN	Recurrent Neural Networks
LSTM	Long Short-Term Memory
KL	Kullback-Leibler divergence
NNCLR	Nearest Neighbor Contrastive Learning Representation
PC	Pancreatic Cancer
PET	Positron Emission Tomography
HDS	Hard hierarchical Decision Structure
SDS	Soft hierarchical Decision Structure
GEO	Gene Expression Omnibus
GNB	Gaussian Naive Bayes
DT	Decision Tree
LR	Logistic Regression
OC	Ovarian Cancer
HGSC	High Grade Serous Carcinoma
SHAP	SHapley Additive exPlanations
ADF	Assumed Density Filtering
NLL	Negative Log Likelihood
BS	Brier Score
RL	Reinforcement Learning

Acknowledgments

I would like to take this opportunity to express my sincere gratitude to my supervisor, Dr. Bala Natarajan, whose guidance, support, and encouragement has been the driving force behind my research. His ability to advise and inspire is truly exceptional. I am very thankful to him for always being available in spite of his busy schedules and for his keen interest in my research work. I would like to extend my thanks to my committee members, Dr. Punit Prakash, Dr. Sanjoy Das and Dr. Michael Higgins, for their time and comments to improve the quality of my research. I would also like to thank my colleagues Dr. Sai Munikoti, Dr. Laya Das, Aabila Tharzeen and Sriram Anbalagan for working alongside and providing constant motivation. I would like to convey my regards to collaborators, Dr. Stefan H Bossmann and Dr. Obdulia Covarrubias-Zambrano from The University of Kansas Medical Center, Dr. Bartosz Szczesny, Dr. Gracie Vargas, Dr. Zifeng (Tony) Tang and Christina Payne from The University of Texas Medical Branch for supporting my research endeavours.

I would like to extend warm regards to my mentor, Dr. Ashwini Wankhede, for his relentless support and motivation during the course of my thesis writing. I would like to thank other members of the CPSWin Research Group, Dr. Rahul Madbhavi, Dr. Amulya Sreejith, Biswajeet Rout, Shweta Dahale, Mohammad Abujubbeh, Dylan Wheeler, Prudhviraj Dhanapala and Priyanka Gautam for their support, encouragement and guidance. Finally, I would like to thank my amazing friends in Kansas, Vidya, Sanket, Ajmal, Nitin, Ruchi and Manoj for their support and great care during the course of my doctoral studies.

This material is based upon work supported by National Science Foundation (NSF) Emerging Frontiers in Research and Innovation program (EFRI) under award number 2129617.

Dedication

This thesis is dedicated to my parents Mr. Muralidhar Agarwala and Mrs. Savita Agarwal for their unconditional love, support and encouragement during my education.

Preface

This dissertation, "Learning and Inferencing Challenges in Human-In-The-Loop Decision Systems," is submitted for the degree of Doctor of Philosophy in the Mike Weigers Department of Electrical and Computer Engineering at Kansas State University. The research was conducted under the supervision of Professor Balasubramaniam Natarajan. This work is original to the best of my knowledge, except where acknowledgments and references are indicated. Most of the work has been presented in the following published peer-reviewed journals and conferences:

1. **Agarwal, D.**, Srivastava, P., Martin-del-Campo, S., Natarajan, B., & Srinivasan, B. (2021, June). Addressing uncertainties within active learning for industrial IoT. In *2021 IEEE 7th World Forum on Internet of Things (WF-IoT)* (pp. 557-562). IEEE.

2. **Agarwal, D.**, Srivastava, P., Martin-del-Campo, S., Natarajan, B., & Srinivasan, B. (2021). Addressing practical challenges in active learning via a hybrid query strategy. *arXiv preprint arXiv:2110.03785* (under review in *International Journal of Approximate Reasoning*).

3. **Agarwal, D.**, & Natarajan, B. (2022, August). Active Learning for Node Classification using a Convex Optimization approach. In *2022 IEEE Eighth International Conference on Big Data Computing Service and Applications (BigDataService)* (pp. 96-102). IEEE.

4. **Agarwal, D.**, Covarrubias-Zambrano, O., Bossmann, S., & Natarajan, B. (2022, February). Impacts of behavioral biases on active learning strategies. In *2022 International Conference on Artificial Intelligence in Information and Communication (ICAIIC)* (pp. 256-261). IEEE.

5. **Agarwal, D.**, & Natarajan, B. (2023). Tracking and handling behavioral biases in active learning frameworks. *Information Sciences*, 641, 119117.

6. Munikoti, S., **Agarwal, D.**, Das, L., Halappanavar, M., & Natarajan, B. (2023). Challenges and opportunities in deep reinforcement learning with graph neural networks: A comprehensive review of algorithms and applications. *IEEE Transactions on Neural Networks and Learning Systems*.

7. **Agarwal, D.**, Covarrubias-Zambrano, O., Bossmann, S. H., & Natarajan, B. (2022). Early detection of pancreatic cancers using liquid biopsies and hierarchical decision structure. *IEEE Journal of Translational Engineering in Health and Medicine*, 10, 1-8.

8. Covarrubias-Zambrano, O., **Agarwal, D.**, Kalubowilage, M., Ehsan, S., Yapa, A. S., Covarrubias, J., Kasi, A., Natarajan, B., & Bossmann, S. H. (2022). A New Hope for Liquid Biopsies: Early Detection of Pancreatic Cancer By Means of Protease Activity Detection in Serum Applying a Hierarchical Decision Structure. *medRxiv*, 2022-10 (under review).

9. Kasi, A., Sun, W., Ehsan, S., Covarrubias, J., Eschliman, K., Neri, R., **Agarwal, D.**, Covarrubias-Zambrano, O., Bossmann, S. H., & Natarajan, B. (2021). Early detection of pancreatic cancers by novel nanobiosensor-based protease biomarkers using hierarchical decision structure.

10. Tang, Z.T., DeJesus, J.E., **Agarwal, D.**, Tharzeen, A., Zhang, Y., Patrikeev, I., Natarajan, B., Bucchieri., F., Zhao, Y., Micci, M.A., Bossmann, S.H., Motamedi, M., Szczesny, B. Circulating cell-free DNA and novel protein biomarkers within extracellular vesicles for the identification of traumatic brain injury severity. *Journal of Neurotrauma*, (Abstract, 2023).

11. Tang, T. Z., Zhao, Y., **Agarwal, D.**, Tharzeen, A., Patrikeev, I., Zhang, Y., DeJesus, J., Bossmann, S. H., Natarajan, B., Motamedi, M. & Szczesny, B. (2024). Serum

Amyloid A and Mitochondrial DNA in Extracellular Vesicles are Novel Markers for Detecting Traumatic Brain Injury in a Mouse Model. *Iscience*.

12. Anbalagan, S., **Agarwal, D.**, Natarajan, B., & Srinivasan, B. (2023, December). Foundational models for fault diagnosis of electrical motors. In *2023 IEEE International Conference on Power Electronics, Smart Grid, and Renewable Energy (PESGRE)* (pp. 1-6). IEEE.

13. Anbalagan, S., GP, S. S., **Agarwal, D.**, Natarajan, B., & Srinivasan, B. (2023). Active Foundational Models for Fault Diagnosis of Electrical Motors. *arXiv preprint arXiv:2311.15516* (under review in *Engineering Applications of Artificial Intelligence*).

14. Munikoti, S., **Agarwal, D.**, Das, L., & Natarajan, B. (2023). A general framework for quantifying aleatoric and epistemic uncertainty in graph neural networks. *Neurocomputing*, 521, 1-10.

15. **Agarwal, D.**, & Natarajan, B. (2024). Uncertainty-guided Active Learning with theoretical bounds on label complexity. (under review in *ACM Transactions on Probabilistic Machine Learning*).

Chapter 1

Introduction

Human-in-the-loop (HITL) decision systems encompass a set of approaches that combine machine and human intelligence for Artificial Intelligence (AI)-enabled applications. The primary goal of such systems is to assist human tasks with Machine Learning (ML) in order to improve their overall efficiency and maximize accuracy [2]. It supports systematic integration of domain knowledge in ML methodologies and allows training of precise prediction models at minimal cost by incorporating human knowledge and experience. Such systems are highly prevalent in a wide variety of application domains [3] including robotics, medical image analysis, smart manufacturing, construction automation, computer vision tasks like person re-identification, natural language processing tasks like entity extraction, entity linking, named entity recognition, question-answering systems and reading comprehension tasks. These synergistic frameworks improve the decision-making process and make AI-enabled systems more robust, accurate and smarter by promoting fusion of machine intelligence with human knowledge and experience. The emergence of AI-based computational models, controllers and decision support engines in the recent years have improved the efficiency and cost effectiveness of such systems and processes. In the context of decision making, HITL systems typically involve a collaborative decision environment, comprising of AI-based models and human experts. The modern semi-supervised ML frameworks like Active Learning (AL) can be well represented as HITL system consisting of human and AI agents.

AL is a form of semi-supervised ML approaches where the learning algorithm leverages information from external sources in order to predict labels for the unlabeled instances in the dataset. The primary motive is to accomplish a higher prediction accuracy with fewer labelled instances as compared to traditional supervised ML methodologies. It has proved to be advantageous in modern ML frameworks involving expensive or wearisome labelling procedures [4]. The learning algorithm in AL settings is referred to as *Active Learner* and the external information source is termed as the *Oracle*. The AL framework can be represented as a collaborative decision environment comprising of Artificial Intelligence (AI) engine, in the form of ML-based classification/regression models; and human experts, in the form of Oracle (i.e., a domain expert). Typically, in such environments, the aim is to learn an efficient decision model by combining domain expertise of the human expert and computational capabilities of the AI model. Recently, AL frameworks are being extensively used for diverse applications including network traffic classification, biomedical text mining, subgraph matching, structural reliability analysis, industrial applications and computer vision tasks. However, the collaboration of AI engines and human experts in a decision environment is not well studied. Both human and AI components of the collaborative decision framework are subject to anomalies and biases. The inadequacy of AI-based algorithms to accommodate for variations in data from different population subgroups gives rise to algorithmic biases like correlation fallacy, selection bias and sampling bias [5]. On the contrary, the uncertainties associated with human decisions result in behavioral biases such as hot hand fallacy, overconfidence, regret aversion bias, representative bias and cognitive bias [6].

Although Deep Neural Networks (DNN)-based decision models have demonstrated remarkable advancements across diverse domains in the recent years, these data-driven models, typically characterized as "black-box" models often encounter failures attributable to various factors. A primary limitation of these decision models is their propensity to overfit, resulting in sub-optimal generalization performance when extrapolating beyond the confines of the available data, particularly in out-of-sample prediction tasks. The predictions generated by neural network models may exhibit physical inconsistencies, thereby reducing their interpretability. One approach to address this challenge involves incorporating prior domain

knowledge, which encapsulates a high-level abstraction of the underlying natural phenomena. The "black-box" models can be made more robust by incorporating additional biases. This issue also prompted the development of explainable artificial intelligence (XAI), which supports AI systems that can explain their internal processes and decision-making methods [7].

Another key challenge that limits the adaptability and reliability of ML methods is their inability to quantify uncertainties in the model predictions. Uncertainty in measured quantities and imprecise information about the underlying structure and features of data and/or model parameters can pose a serious impediment to the efficiency of the learning process and quality of the resulting decision models. Uncertainty in estimated parameters and structure of trained model is a fundamental modeling challenge that imposes restrictions on the confidence of predictions. Learning representations in an uncertainty-aware manner is fundamental to producing robust models and reliable predictions. Models that do not account for these sources of uncertainty can be over-confident in their predictions. Moreover, neural network models are often prone to overfitting that limits their ability to generalise [8]. These factors can pose serious problems to effective utilization of the available information in the model building process as well as reliable interpretation of the model predictions under adverse situations. Furthermore, the knowledge of uncertainty quantification is also leveraged to modify training objectives and thereby, improve model performance and robustness.

Given the wide range of applications supported by HITL decision systems and their ability to enhance the overall efficiency of ML models by integrating human knowledge, there exist challenges at multiple levels. The learning and inferencing challenges in HITL decision systems can be broadly classified into three levels, as illustrated in Figure 1.1. Each of these category of challenges is discussed next.

1. ***Representation-Level Challenges***: The challenges at the representation level are primarily focused upon formulating and representing the components of HITL systems in an effective manner. This would support a methodical study and analysis of the synergistic human-AI collaborative decision environment adequately.

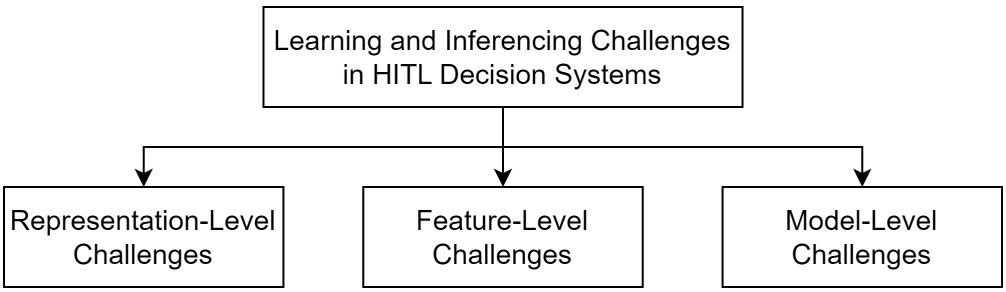


Figure 1.1: Categorization of learning and inferencing challenges in HITL decision systems.

2. ***Feature-Level Challenges***: The challenges at the feature level are directed towards designing appropriate feature engineering steps in order to improve representation and learning of the decision systems. Efficient feature engineering also helps to address the issue of lack of explainability of the decision models.

3. ***Model-Level Challenges***: The challenges at the model level are aimed at improving the robustness and predictive ability of the decision models. This is done by quantifying uncertainties from various sources within the learning scheme. The knowledge of uncertainty is then leveraged to modify model training objectives and thereby, improve model performance and robustness.

Based on these perspectives, this dissertation seeks to address a few fundamental research questions. These research questions and contributions of the dissertation that aim to address them are discussed in the following sections.

1.1 Research Questions

In order to address learning and inferencing challenges in HITL decision systems, the following research questions are formulated:

Question 1: *What are the practical challenges involved in implementation of semi-supervised learning approaches like AL, in the context of HITL decision systems?*

Question 2: *With regard to the human component in AL methodologies:*

a) *What are the impacts of different types of behavioral biases on AL strategies?*

b) *How to model, track and subsequently integrate the behavioral biases within the learning scheme, so that the resulting decision models are more robust and accurate?*

Question 3: *Can the generalizability of HITL decision models be improved by incorporating observational biases within the learning frameworks?*

Question 4: *How to make the predictions of a decision model "explainable", so that they can serve as effective decision support tools in safety-critical applications like medical diagnosis?*

Question 5: *How can the uncertainty quantification be leveraged to modify training objectives, specifically to guide the querying process in the context of AL?*

1.2 Contributions

To address Research Question 1, Chapter 3 of this dissertation presents a novel hybrid query strategy-based AL framework that addresses three practical challenges, namely, cold-start, oracle uncertainty and performance evaluation of Active Learner in the absence of ground truth. The idea of AL is extended to representation learning in non-Euclidean space like graphs in Chapter 4. An AL framework is proposed to address node classification task in attributed graphs using a convex optimization approach. The proposed method integrates both topological information as well as node attributes within the learning scheme. The key contributions of these chapters are as follows:

- The cold-start problem is addressed using pre-clustering to assign labels to instances close to the cluster centroids.
- The issue of oracle uncertainty in handled by making a combined decision based on the confidence scores of model predictions and those of the human experts.
- A set of metrics derived from Active Learning heuristics is proposed to evaluate the quality of the labels. These metrics are also used to identify the instance that requires switching to a different query algorithm under the proposed hybrid query strategy.

- Four graph-theoretic metrics are used as AL heuristics/acquisition functions for active selection of nodes in graphs.

- Both node attributes and topological information are incorporated in the learning scheme. The node features are exploited while training the GNN-based decision model and topological information is considered during selective sampling of the nodes.

- An inductive GNN approach is used for building node classification models. This allows the approach to be scalable across graphs of different sizes as well as subgraphs within the given graph.

More details on the proposed methodologies for handling practical AL challenges and corresponding experimental evaluation can be found in Chapters 3 and 4, and in the following articles:

- Agarwal, D., Srivastava, P., Martin-del-Campo, S., Natarajan, B., & Srinivasan, B. (2021, June). Addressing uncertainties within active learning for industrial IoT. In *2021 IEEE 7th World Forum on Internet of Things (WF-IoT)* (pp. 557-562). IEEE.

- Agarwal, D., Srivastava, P., Martin-del-Campo, S., Natarajan, B., & Srinivasan, B. (2021). Addressing practical challenges in active learning via a hybrid query strategy. *arXiv preprint arXiv:2110.03785* (under review in *International Journal of Approximate Reasoning*).

- Agarwal, D., & Natarajan, B. (2022, August). Active Learning for Node Classification using a Convex Optimization approach. In *2022 IEEE Eighth International Conference on Big Data Computing Service and Applications (BigDataService)* (pp. 96-102). IEEE.

The impact of different behavioral biases is demonstrated in an AL context to address Research Question 2a. Chapter 5 of this dissertation represents AL as a collaborative decision environment consisting of AI engines (ML-based models) and human experts (Oracle). The behavioral biases are simulated by providing pre-engineered human decisions during the

input steps. Chapter 6 further addresses Research Question 2b by presenting a systematic framework for modeling, tracking and adaptation of behavioral biases in human-AI interactive decision environments within the context of AL. The specific contributions are as follows:

- The human-AI interactive decision environment is modeled in the form of a coupled dynamical system consisting of human experts and AI engines using communication graphs. This novel modeling approach helps characterize the dynamics of error variables that arise when the communication graph changes over iterative queries within the AL environment.

- For the first time, a novel Exploitation-Scrutinization-Investigation-Scrutinization (ESIS) strategy is proposed for tracking of behavioral biases within AL settings. Specifically, the querying process is strategically designed and split into four iterative stages to track the extent of behavioral biases and subsequently infer the grade of human expert.

- Once the behavioral biases are tracked, a unique model adaptation framework is introduced to handle these biases by assigning appropriate weights to samples during the iterative training procedure in AL. These weights are evaluated based on the values of expected collaborative utility function. The samples with higher value of expected collaborative utility are allotted higher weights during training and vice-versa.

More details on simulation, modeling, tracking and model adaptation of behavioral biases can be found in Chapters 5 and 6, and in the following articles:

- Agarwal, D., Covarrubias-Zambrano, O., Bossmann, S., & Natarajan, B. (2022, February). Impacts of behavioral biases on active learning strategies. In *2022 International Conference on Artificial Intelligence in Information and Communication (ICAIIC)* (pp. 256-261). IEEE.

- Agarwal, D., & Natarajan, B. (2023). Tracking and handling behavioral biases in active learning frameworks. *Information Sciences*, 641, 119117.

To address Research Question 3, Chapter 7 presents two case studies, each demonstrating a way to incorporate observational biases within the learning frameworks.

- The first case study illustrates a GNN-based framework developed for identification of potential biomarkers for biological phenomena. This involves incorporating knowledge from an existing database to develop a graph that incorporates protein-protein interactions as edge connections and is used as input data to identify the potential biomarkers.

- The second case study presents a foundational model-based AL framework for fault diagnosis of electrical motors, that utilizes less amount of labeled samples, which are most informative and harnesses a large amount of available unlabeled data by effectively combining AL and Contrastive Self Supervised Learning (SSL) techniques. This model consists of two key components: the construction of the backbone model, followed by the fine-tuning procedure for various target tasks using less amount of labeled samples.

More details related to these case studies can be found in Chapter 7, and in the following articles:

- Tang, Z.T., DeJesus, J.E., Agarwal, D., Tharzeen, A., Zhang, Y., Patrikeev, I., Natarajan, B., Bucchieri., F., Zhao, Y., Micci, M.A., Bossmann, S.H., Motamedi, M., Szczesny, B. Circulating cell-free DNA and novel protein biomarkers within extracellular vesicles for the identification of traumatic brain injury severity. *Journal of Neurotrauma*, (Abstract, 2023).

- Tang, T. Z., Zhao, Y., Agarwal, D., Tharzeen, A., Patrikeev, I., Zhang, Y., DeJesus, J., Bossmann, S. H., Natarajan, B., Motamedi, M. & Szczesny, B. (2024). Serum Amyloid A and Mitochondrial DNA in Extracellular Vesicles are Novel Markers for Detecting Traumatic Brain Injury in a Mouse Model. *Iscience*.

- Anbalagan, S., Agarwal, D., Natarajan, B., & Srinivasan, B. (2023, December). Foundational models for fault diagnosis of electrical motors. In *2023 IEEE International*

Conference on Power Electronics, Smart Grid, and Renewable Energy (PESGRE) (pp. 1-6). IEEE.

- Anbalagan, S., GP, S. S., Agarwal, D., Natarajan, B., & Srinivasan, B. (2023). Active Foundational Models for Fault Diagnosis of Electrical Motors. *arXiv preprint arXiv:2311.15516* (under review in *Engineering Applications of Artificial Intelligence*).

To address Research Question 4, Chapter 8 presents two case studies to demonstrate different ways of explaining the predictions made by decision models.

- The soft hierarchical decision structure proposed for early-stage detection of pancreatic cancer provides confidences associated with the predicted labels in the form of probability values for each class. The differences between these probability values provide an indication of confidence associated with the corresponding prediction. It helps the doctors to determine whether additional tests are required for proper diagnosis.

- The active explainable decision framework proposed for early-stage detection of ovarian cancer is supported by feature relevance explanation technique. In this framework, the feature relevance scores explain as to how much each biomarker is contributing to arrive at a prediction for the sample under consideration.

More details related to these case studies can be found in Chapter 8, and in the following articles:

- Agarwal, D., Covarrubias-Zambrano, O., Bossmann, S. H., & Natarajan, B. (2022). Early detection of pancreatic cancers using liquid biopsies and hierarchical decision structure. *IEEE Journal of Translational Engineering in Health and Medicine*, 10, 1-8.

- Covarrubias-Zambrano, O., Agarwal, D., Kalubowilage, M., Ehsan, S., Yapa, A. S., Covarrubias, J., Kasi, A., Natarajan, B., & Bossmann, S. H. (2022). A New Hope for Liquid Biopsies: Early Detection of Pancreatic Cancer By Means of Protease Activity Detection in Serum Applying a Hierarchical Decision Structure. *medRxiv*, 2022-10 (under review).

- Kasi, A., Sun, W., Ehsan, S., Covarrubias, J., Eschliman, K., Neri, R., Agarwal, D., Covarrubias-Zambrano, O., Bossmann, S. H., & Natarajan, B. (2021). Early detection of pancreatic cancers by novel nanobiosensor-based protease biomarkers using hierarchical decision structure.

To address Research Question 5, Chapter 9 proposes a novel framework to leverage the knowledge of uncertainty to guide querying process in AL. Given a neural network-based decision model and a tolerable threshold on the uncertainty of its predictions, the prediction uncertainty is first propagated back to the input side using the Assumed Density Filtering (ADF) approach. This knowledge is then leveraged to guide the informative sample selection via querying in AL. The lower and upper bounds for label complexity are also derived in the context of AL approach considered in this work. The proposed methodology is generic and can be applied to any AL setup, irrespective of the query strategy and base decision model.

More details on the analytical results along with their proofs and experimental evaluation can be found in Chapter 9, and in the following articles:

- Munikoti, S., Agarwal, D., Das, L., & Natarajan, B. (2023). A general framework for quantifying aleatoric and epistemic uncertainty in graph neural networks. *Neurocomputing*, 521, 1-10.

- Agarwal, D., & Natarajan, B. (2024). Uncertainty-guided Active Learning with theoretical bounds on label complexity. (under review in *ACM Transactions on Probabilistic Machine Learning*).

1.3 Organization of the Dissertation

The remainder of this dissertation is organized as follows. Chapter 2 provides a background on several key concepts that are needed to develop and understand novel methodologies proposed in this dissertation. Chapter 3 presents a novel hybrid query strategy-based AL framework that addresses three practical challenges, namely, cold-start, oracle uncertainty

and performance evaluation of Active Learner in the absence of ground truth. The idea of AL is extended to representation learning in non-Euclidean space like graphs in Chapter 4. Chapter 5 represents AL as a collaborative decision environment consisting of AI engines (ML-based models) and human experts (Oracle). The behavioral biases are simulated by providing pre-engineered human decisions during the input steps. Chapter 6 further presents a systematic framework for modeling, tracking and adaptation of behavioral biases in human-AI interactive decision environments within the context of AL. Chapter 7 presents two case studies, each demonstrating a way to incorporate observational biases within the learning frameworks. Other two case studies to demonstrate different ways of explaining the predictions made by decision models are illustrated in Chapter 8. Chapter 9 proposes a novel framework to leverage the knowledge of uncertainty to guide querying process in AL. Finally, the key conclusions of this dissertation are presented and avenues for future research are suggested in Chapter 10.

Chapter 2

Background

This chapter provides a background on several key concepts that are needed to develop and understand novel methodologies proposed in this dissertation.

2.1 Active Learning

Active Learning (AL) is a powerful tool to address modern Machine Learning (ML) problems with significantly fewer labeled training instances. AL comprises of ML techniques where the learning algorithm is allowed to communicate with external information sources and collect labels corresponding to unlabeled instances in the dataset. It allows the learning algorithm to leverage information from external sources (e.g., human experts) and train an efficient decision model using much fewer labeled instances as compared to traditional supervised ML methodologies. This is highly beneficial in critical applications where labeling process is time-consuming or expensive [9]. In AL settings, the external source of information is referred to as *Oracle* and the learning algorithm is termed as *Active Learner*. The Active Learner is permitted to select the data it wants to learn from. Thus, it poses queries to the Oracle in the form of unlabeled data instances, and the Oracle is expected to supply corresponding labels. Combining the capabilities of AI engine (as ML-based decision models) and human experts (as Oracle), AL frameworks can be portrayed as a human-AI collaborative

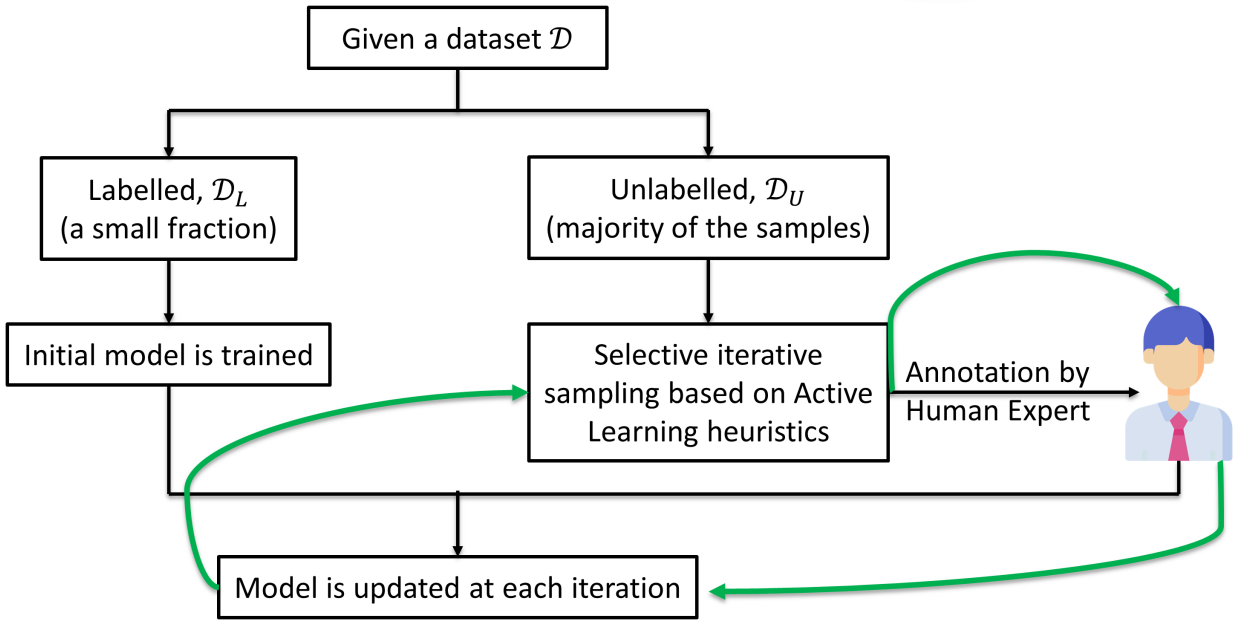


Figure 2.1: Overall strategy of a typical AL framework.

decision environment. The fundamental motive of such learning frameworks is to combine the computational capabilities of AI engine with domain expertise of human experts in order to learn an accurate decision model in an efficient manner.

The overall strategy of a typical AL framework is described in Figure 2.1 [10]. For any dataset $\mathcal{D}$, a majority of the samples are unlabeled ($\mathcal{D}_U$) and only a small fraction of them are labeled ($\mathcal{D}_L$). The primary objective is to reliably predict labels for all the unlabeled samples using significantly fewer labeled cases for training than traditional supervised machine learning frameworks. This is achieved by allowing the *Active Learner* to select the data it wishes to learn from, i.e., presenting specific unlabeled instances as *queries* to the *Oracle* and obtaining the appropriate labels. AL commences by training an initial decision model using $\mathcal{D}_L$. Further, the relevant AL heuristics, such as classifier uncertainty, entropy of class probabilities and margin uncertainty are optimized to select queries in an iterative manner. At each query step, the model is retrained by including the query along with its label in $\mathcal{D}_L$.

This interactive modeling procedure can be well represented as a collaborative decision environment between AI engine (ML model) and human expert (Oracle, i.e., domain expert).

2.2 Graph Neural Networks

Recently, there has been a surge of interest in learning with graph-structured data, like, biological, financial and social networks, and knowledge graphs. There are several advantages of representing data in form of graphs. Firstly, they provide an intuitive approach to visualize concepts of interactions and relationships. Additionally, it allows us to elucidate a complex problem by transforming it into simpler representations. The concepts of graph theory can be exploited to model and analyze such forms of data representations. However, it is difficult to analyze and interpret graph-structured data using traditional Deep Neural Network (DNN)-based learning algorithms. This is primarily because graphs cannot be described in Euclidean space, which makes it difficult to interpret as compared to other types of data like images or text. Furthermore, the irregular structure of graphs, inconsistent size of unordered nodes and variable neighborhood configuration of the nodes prevents crucial mathematical operations like convolutions to be applied on graph-structured data. GNN help to overcome these limitations by extending DNN architectures for graph domain.

GNNs are typically based on recursive message passing (i.e., neighborhood aggregation), wherein every node in the graph aggregates attributes of its neighbors in order to compute its own feature vector [11, 12]. This allows a node to be represented by its transformed feature vector or node embedding after a specific number of aggregation iterations, thereby capturing the structural information across its p-hop neighborhood. Several variants of GNN with diverse neighborhood aggregation functions and different pooling schemes have been presented in the literature [11, 13–27]. These GNN methodologies have been empirically shown to achieve state-of-the-art performance on many downstream tasks such as link prediction, node classification and graph classification. In this dissertation, GraphSAGE [17] is used as an inductive node embedding approach that concurrently learns both the topological structure and distribution of features for a node in its local neighborhood. The operation executed at i^{th} node embedding layer is given by eq. (2.1).

$$
\begin{aligned}
h_u^{(i)} &= f^{(i)}\left(h_u^{(i-1)}, h_{N(u)}^{(i-1)}\right) \\
&= g\left[\theta_C^{(i)} h_u^{(i-1)} + \theta_A^{(i)} \tilde{A}\left(h_{N(u)}^{(i-1)}\right)\right]
\end{aligned}
\tag{2.1}
$$

Here, $h_u^{(i)}$ represents the node embedding of node u at i^{th} layer; $\tilde{A}$ denotes the aggregation operation; θ_C and θ_A are the parameters of the combination and aggregation operation of GNN, respectively; $N(u)$ describes the neighborhood of node u and $g[\cdot]$ denotes the activation function.

2.3 Active Learning via Convex Optimization

AL methodology can be illustrated as a framework based on convex programming [28] that leverages Dissimilarity-based Sparse Modeling Representative Selection (DSMRS) algorithm [29, 30] for selecting most informative nodes in the given graph-structured data. There are numerous advantages of using a convex optimization-based AL framework as compared to the traditional iterative querying mechanism [28]. Firstly, it incorporates aspects of both uncertainty sampling as well as sample diversity while formulating the convex optimization problem. This allows to select multiple samples from the dataset which are not only informative but are also diverse with respect to each other. Moreover, the data distribution is assimilated via a dissimilarity matrix in the problem formulation. The formulation of AL as a convex optimization problem is discussed next.

Given a graph G with N nodes and dissimilarity matrix $\boldsymbol{D} = \{d_{ij}\}_{i,j=1,2,3,\ldots,N}$, the goal is to find a small subset of nodes that are well representative of the entire graph. Here, d_{ij} represents how well the node j is represented by another node i. The smaller values of d_{ij} denotes better representation and vice-versa. The dissimilarity values are assumed to be non-negative and $d_{jj} \leq d_{ij}$ for all i, j. In order to find the representative nodes of the graph, $\boldsymbol{Z} = \{z_{ij}\}_{i,j=1,2,3,\ldots,N}$ is introduced as an optimization variable. $z_{ij} \in [0, 1]$ is associated to d_{ij}

and indicates the probability of node i being a representative of node j. The optimization function consists of two components: (i) encoding cost of all N nodes using the representative nodes, and (ii) penalizing the number of selected representatives. The encoding cost of node j via i can be expressed as $d_{ij}z_{ij} \in [0, d_{ij}]$. So, the total encoding cost for all the nodes in the graph is given by eq. (2.2).

$$\sum_{i,j} d_{ij}z_{ij} = \text{tr}(\boldsymbol{D}^T\boldsymbol{Z}) \tag{2.2}$$

The number of representative nodes can be directly associated to the the number of non-zero rows in $\boldsymbol{Z}$ [29]. Consequently, a convex surrogate for the cost related to number of selected representatives is given by eq. (2.3), where $q \in \{2, \infty\}$.

$$\sum_{i=1}^{N} ||z_i||_q \triangleq ||\boldsymbol{Z}||_{q,1} \tag{2.3}$$

Combining both the components in eqs. (2.2) and (2.3) together, the overall convex optimization problem is given in eq. (2.4).

$$\min \lambda||\boldsymbol{Z}||_{q,1} + \text{tr}(\boldsymbol{D}^T\boldsymbol{Z}) \text{ s.t. } \boldsymbol{Z} \geq 0, \mathbf{1}^T\boldsymbol{Z} = \mathbf{1}^T \tag{2.4}$$

Here, the λ parameter balances the costs associated with encoding and number of representatives, and controls the batch size in AL. A smaller value of λ lays more importance on better encoding, thereby obtaining more representative nodes. On the other hand, a higher value of λ corresponds to more emphasis on penalizing, thereby obtaining less representative nodes. The constraints ensure that each column of $\boldsymbol{Z}$ is a probability vector.

This DSMRS-based problem formulation can be well extended to AL by incorporating sample informativeness score and sample diversity score. The informativeness score for a node u, $s_{inf}(u) \in [0, 1]$, represents the degree of its importance or informativeness. A higher value of $s_{inf}(u)$ signifies that that node u is highly representative of the given graph. On the other hand, a lower value of $s_{inf}(u)$ represents that node u is not so representative and conveys less information about the given graph. The diversity score for a node u is given by

eq. (2.5).

$$s_{div}(u) = \frac{\min_{j \in \mathcal{L}} d_{ju}}{\max_{k \in \mathcal{U}} \min_{j \in \mathcal{L}} d_{jk}} \tag{2.5}$$

Here, $\mathcal{L}$ and $\mathcal{U}$ are the sets of indices of labeled and unlabeled nodes respectively.

If the closest labeled node to an unlabeled node u is very similar to it, $s_{div}(u) \to 0$, and selecting such a node doesn't promotes diversity. On the contrary, when all the labeled nodes are very dsisimilar from an unlabeled node u, $s_{div}(u) \to 1$, and selecting such a node for labeling would be beneficial as it would be different from the other labeled nodes. The combined score matrix ($\boldsymbol{S}$) is given by eq. (2.6), where the overall score for each node $s(u_i)$ is evaluated as shown in eq. (2.7).

$$\boldsymbol{S} = \text{diag}\left(s(u_1), s(u_2), s(u_3), ..., s(u_{|\mathcal{U}|})\right) \tag{2.6}$$

$$s(u_i) = \max\{s_{inf}(u_i), s_{div}(u_i)\} \tag{2.7}$$

Other mathematical operations like mean or minimum can be used instead of max to compute the overall score for each node in eq. (2.7). Finally, AL process can be represented in the form of an optimization problem as shown in eq. (2.8).

$$\min \lambda ||\boldsymbol{S}\boldsymbol{Z}||_{q,1} + \text{tr}(\boldsymbol{D}^T \boldsymbol{Z}) \text{ s.t. } \boldsymbol{Z} \geq 0, \mathbf{1}^T \boldsymbol{Z} = \mathbf{1}^T \tag{2.8}$$

If an unlabeled node u_i has lower score $s(u_i)$, the optimization program puts less penalty on the i^{th} row of $\boldsymbol{Z}$ being non-zero, and vice-versa. The convexity of the optimization problem (2.8) is analyzed in Chapter 4.

2.4 Fault Diagnosis of Electrical Motors

An electrical motor comprisess of mechanical elements such as the stator, rotor, bearings and electrical components like windings and end rings. They are designed to operate across

various industrial and environmental conditions, subjecting them to diverse forms of stress. These stress factors lead to different types of faults within the electrical motor, broadly categorized as mechanical or electrical faults. Among the faults that afflict electrical motors, bearing faults, shaft misalignment, rotor imbalance, and inter-turn short circuit faults are most prominent. Existing literature indicates that most electrical motor breakdowns can be attributed to mechanical faults [31]. In this work, the primary mechanical faults commonly observed in electrical motors are considered. This includes bearing faults of various sizes occurring at different locations (such as inner-race, ball, and outer-race) and shaft misalignment faults exhibiting varying degrees of severity.

2.5 Transformers

The multi-head self-attention mechanism of the transformer network enables it to capture better the complex dynamics of input data. The key components of transformer architecture are described as follows:

2.5.1 Multi-head Self-Attention

The Multi-head Self-Attention (MSA) gives the transformer greater power to encode multiple relationships and nuances for each input sample [32]. A mapping from a query (Q) and a collection of key-value pairs (K) to an output (V), where Q, K, and V are all vectors, can be thought of as an attention mechanism. The weight matrix (Attention Distribution, AD) is derived from Q and K. The result is a weighted sum over V. Characterizing sequence similarity is the essence of attention distribution. The attention relationship is mainly calculated by the scaled dot-product attention method. Given a set of input matrices, $Q \in \mathbb{R}^{m\times n}, K \in \mathbb{R}^{m\times n}$, and $V \in \mathbb{R}^{m\times n}$, the mathematical expression of self-attention calculation is given as follows [33]:

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{n}}\right)V \tag{2.9}$$

where $\sqrt{n}$ is a scale factor.

The multi-head self-attention enables parallel computation of self-attention by getting multiple sets of Q, K, and V matrices in different initialization forms. The concurrent computation of several sets of attention relations is then translated into a set of outputs using a transformation matrix W^0. It can be expressed as [33]:

$$\begin{aligned} \text{MultiHead}(Q, K, V) &= Concat(head_1, ..., head_i)W^o \\ head_i &= \text{Attention}(Q\text{W}_i^Q, K\text{W}_i^K, V\text{W}_i^V) \end{aligned} \tag{2.10}$$

2.5.2 Transformer Block

The transformer block includes Layer Normalization (LN), Feed-forward Neural Network (FNN), and Residual Connections (RC) along with the above discussed MSA layer. A transformer block is the composition of transformer encoders and transformer decoders. Given an input of x $\in \mathbb{R}^{m \times n}$, the mathematical expression of LN is given as follows:

$$\text{LN}(x) = \frac{x - \mu}{\sqrt{\sigma^2 + \varepsilon}} \cdot g + b \tag{2.11}$$

where, μ and σ^2 represent the mean and variance of x. g and b are scaling parameters and translation parameters, respectively; and ε generally takes the value of $1e^{-5}$.

The attention relations calculated parallel in the FNN can be mathematically expressed as follows:

$$\text{FNN}(x) = (\text{GeLU}(x)\ W_1 + b_1)\ W_2 + b_2 \tag{2.12}$$

where, GeLU (Gaussian error Linear Unit) is a nonlinear activation function, W_1 and W_2 indicate weights, and b_1 and b_2 indicate biases. Considering x as input, $f(x)$ is the output of the weight layer. The calculation of residual connections is expressed as follows:

$$\text{RC}(x) = f(x) + x. \tag{2.13}$$

Hence, for a transformer block, the output x_o of its complete forward propagation process can be represented as [34]:

$$\begin{aligned} \hat{x} &= \mathrm{LN}(\mathrm{MSA}(x_i)) + x_i \\ x_o &= \mathrm{LN}(\mathrm{FNN}(\hat{x})) + \hat{x} \end{aligned} \tag{2.14}$$

2.6 Contrastive Learning

Contrastive Learning is a paradigm of SSL based on the idea that samples of the same class are closer to each other in latent representation space and samples of different classes are farther. Contrastive Learning has found many applications in Computer Vision and Pattern Recognition. Learning in the Contrastive Learning framework is achieved by pushing "similar" looking samples towards each other and "dissimilar" looking samples farther from each other. [35] provides a loss function called *NT-Xent*, which achieves the above-mentioned Learning task via a Contrastive Learning framework. The loss function is given as,

$$l(i,j) = -\log \frac{\exp\left(\mathrm{sim}\left(z_i, z_j\right)/\tau\right)}{\sum_{k=1}^{2N} \mathbb{I}_{[k \neq i]} \exp(\mathrm{sim}\left(z_i, z_k\right)/\tau)} \tag{2.15}$$

where, sim(u, v) is defined as,

$$\mathrm{sim}(u,v) = \frac{\langle u, v \rangle}{\|u\|\|v\|} \tag{2.16}$$

The loss functions employ the projections of the samples i, j onto the latent space, which are z_i, z_j and the similarity between the two projections are represented using $\mathrm{sim}(u, v)$ and τ is the softmax temperature. This similarity metric is used to determine how "similar" i, j are with respect to the other samples in the batch of size $2N$. The loss is high if i, j are more "similar" compared to other samples in the batch and vice versa. The framework aims to learn the representation space where "similar" samples are closer.

2.7 Foundational Models

Transfer learning and scale provide the technical background for foundational models [36]. Transfer learning involves the application of acquired knowledge from one task to another. Pretraining is the predominant approach to implementing transfer learning in the field of deep learning. The model undergoes training on a surrogate task and is subsequently tuned to align with the downstream task of interest. The scalable attribute enhances the potential of the foundational model. In recent times, there has been a notable adoption of foundational models in the domains of Natural Language Processing (NLP) and computer vision. An example of a task in Bidirectional Encoder Representations from Transformers (BERT) [37] is the masked language modelling challenge, which entails predicting a word that is absent from a phrase by considering its surrounding context. Nevertheless, the profound influence of these fundamental models on NLP does not merely stem from their ability to generate text but rather from their exceptional versatility and adaptability. The concept of foundational models has been implemented in our prior work on fault diagnosis of electrical machines [38]. In this work, the idea of foundational models is extended to incorporate AL and contrastive SSL so that the transformer-based backbone network can be trained using fewer labeled instances as compared to the state-of-the-art.

2.8 Label Complexity

Label complexity refers to the number of queries adequate for an AL algorithm to attain a desirable level of predictive performance, (e.g., accuracy score in the context of classification tasks). The label complexity function, generally indicated by Λ, maps a distribution $\mathcal{P}_{XY}$ and the values $\epsilon, \delta \in [0,1]$ to a value given by $\Lambda(\epsilon, \delta, \mathcal{P}_{XY}) \in \mathbb{N} \cup \{\infty\}$ [39]. Here, X represents the feature space and Y represents the label space within the learning scheme. We have the following definition [39]:

Definition 1. *For every $\mathcal{P}_{XY}$ over $X \times Y$, $\epsilon \geq 0$, $\delta \in [0,1]$ and every integer $n \geq \Lambda(\epsilon, \delta, \mathcal{P}_{XY})$, an AL algorithm attains label complexity Λ if it generates a classifier $\hat{h}$ such*

that the error rate $E(\hat{h}) \leq \epsilon$ *with probability at least* $(1-\delta)$.

In general, the primary interest of an AL algorithm lies in label complexity of achieving an error rate that is lower than the best error rate in the hypothesis class (comprising of a fixed set $\mathbb{C}$ of classifiers). Considering noise rate as

$$\nu = inf_{h \in \mathbb{C}} E(h), \tag{2.17}$$

it is assumed that this can be achieved by a classifier $f^* \in \mathbb{C}$, i.e., $E(f^*) = \nu$. Given a sequence of labeled data points $\mathcal{L} \in \bigcup_{m \in \mathbb{N}} (X \times Y)^m$ and a classifier h, the empirical error rate of h with respect to $\mathcal{L}$ corresponds to,

$$E_{\mathcal{L}}(h) = \frac{1}{|\mathcal{L}|} \sum_{(x,y) \in \mathcal{L}} \mathbb{1}[h(x) \neq y] \tag{2.18}$$

where $\mathbb{1}[\cdot]$ is indicator function. Eq. (2.18) represents the fraction of misclassified points in $\mathcal{L}$ by h. Moreover, the version space induced by $\{X_1, X_2, X_3, ..., X_m\}$ is denoted by [39]:

$$V_m^* = \{h \in \mathbb{C} : \forall i \leq m, h(X_i) = f^*(X_i)\} \tag{2.19}$$

Given a set of classifiers $\mathcal{H}$ and an arbitrary value $\epsilon \in [0,1]$, the ϵ-minimal set is defined as $\mathcal{H}(\epsilon) = \{h \in \mathcal{H} : E(h) - inf_{g \in \mathcal{H}} E(g) \leq \epsilon\}$. Additionally, for a classifier h, the ϵ-ball centered at h is given by $B_{\mathcal{H},\mathcal{P}}(h,\epsilon) = \{g \in \mathcal{H} : \mathcal{P}(x : g(x) \neq h(x)) \leq \epsilon\}$. When $\mathcal{H} = \mathbb{C}$, we abbreviate $B_{\mathcal{P}}(h,\epsilon) = B_{\mathbb{C},\mathcal{P}}(h,\epsilon)$. When $\mathcal{P}$ is evident from the problem context, we abbreviate $B_{\mathcal{H}}(h,\epsilon) = B_{\mathcal{H},\mathcal{P}}(h,\epsilon)$ and $B(h,\epsilon) = B_{\mathbb{C},\mathcal{P}}(h,\epsilon)$. Furthermore, the radius of set $\mathcal{H}$ is indicated by $radius(\mathcal{H}) = sup_{h \in \mathcal{H}} \mathcal{P}(x : h(x) \neq f^*(x))$. This also indicates the least value of ϵ for which $B_{\mathcal{H}}(f^*,\epsilon) = \mathcal{H}$. Consequently, the region of disagreement of $\mathcal{H}$ is specified as [39]:

$$DIS(\mathcal{H}) = \{x \in X : \exists h, g \in \mathcal{H} \; s.t. \; h(x) \neq g(x)\} \tag{2.20}$$

One of the aims in this work is to obtain bounds for label complexity using the uncer-

tainty information propagated to the input layer of neural network-based decision models. A suitable way to do this is to bound $\mathcal{P}(DIS(\mathcal{H}))$ by a homogeneous linear function of a bound on $radius(\mathcal{H})$. The coefficient in this linear function is known as the disagreement coefficient, indicated by $\theta(\cdot)$. For any $V \subseteq \mathbb{C}$ and $r > \max\{radius(V), r_0\}$, $\mathcal{P}(DIS(V)) \leq \theta(r_0)r$.

2.9 Uncertainty Quantification

The presence of uncertainties in measured quantities and inaccurate information regarding the underlying structure and features of data and/or model parameters can significantly hinder the effectiveness of the learning process and the accuracy of the resultant decision models. Uncertainty in the estimated parameters and structure of a trained decision model poses a significant modelling issue, since it limits the level of confidence that can be placed on the predictions made by the model. Uncertainty-aware learning of model representations is essential for the development of robust models and reliable predictions. Models that fail to consider these sources of uncertainty may exhibit excessive confidence in their forecasts. Furthermore, it is worth noting that neural network models frequently exhibit a susceptibility to overfitting, which imposes constraints on their capacity to generalise. This might present significant challenges to the efficient use of the existing knowledge during the process of model training, as well as accurate interpretation of the model predictions in unfavourable circumstances. Furthermore, uncertainty quantification of model predictions is also crucial for guiding strategic sampling in the context of AL.

Chapter 3

Uncertainties in Active Learning

Active Learning (AL) constitutes an area of machine learning where the learning algorithm is allowed to interact with external sources of information to obtain labels corresponding to the unlabelled instances in the dataset. AL methods are advantageous in modern machine learning frameworks, where the labelling process may be laborious, long-standing or expensive [40, 41]. However, numerous challenges are encountered during implementation of AL frameworks in practical scenarios due to inherent assumptions [42]. This chapter presents a novel hybrid query strategy-based AL framework that addresses three practical challenges, namely, cold-start, oracle uncertainty and performance evaluation of Active Learner in the absence of ground truth.

3.1 Related Work

A challenge within AL frameworks is the cold-start problem, which occurs due to absence of labels in the beginning for training an initial model. Among the alternatives, there is a proposal to incorporate uncertainty and diversity sampling into a unified process to select most representative samples as the initial labelled dataset [43]. Gong et al. [44] propose a novel inference method based on Bayesian Deep Latent Gaussian Model (BELGAM) to select initial training instances. Another option to address the cold-start problem is an un-

supervised matching method that proposes bootstrapping active learning [45]. Furthermore, Deng et al. [46] introduced a sequence-based adversarial learning model to select initial set of training instances for the AL methods. In this chapter, an AL framework is proposed that employs a pre-clustering step to select the points closest to the centroids of each cluster as instances for initial labelled dataset.

The problem of oracle uncertainty occurs due to the assumption of oracle being infallible in AL methodologies. A majority of approaches proposed in the literature can be classified into two categories. The first category of AL methods [47–49] contemplates a multi-annotator set-up and combines the confidence levels of multiple uncertain annotators to obtain a single confidence metric. The other category of approaches [50–52] accounts for the oracle noise by adding denoising layers or including penalties in objective functions within the querying framework. In contrast to the prior efforts, a generic framework is proposed to handle oracle uncertainty that builds upon the ambiguity surrounding the expertise of the labeler and the confidence in the given labels. The confidence scores associated with the model predictions, as well as the labels supplied by experts, are considered as a part of the querying process. Such a strategy to handle oracle uncertainty by incorporating additional feedback from the experts in the form of confidence levels has not been reported in the Active Learning literature so far to the best of our knowledge and stands out as a novel contribution of this work.

A majority of the AL methodologies report model performance in terms of classification accuracy, which requires ground truth information. However, it is not feasible to be informed about ground truth in practical scenarios. The use of AL heuristics as surrogate metrics is proposed to evaluate the performance of the Active Learner. These metrics are observed to mimic the role of classification accuracy in model evaluation. The AL methodology presented in [53] addresses this issue along with handling oracle uncertainty. However, it still lacks the ability to solve cold-start problem in practical AL scenarios. The proposed AL framework possesses the capability to address multiple challenges simultaneously, i.e., cold-start, oracle uncertainty and performance evaluation of Active Learner in the absence of ground truth. Additionally, it supports the use of hybrid query strategies, rather than employing a single query strategy during the entire querying process. This work demonstrates that the use of

hybrid query strategies helps in reducing the computational cost, while delivering at par or better performance than the individual query strategies. The values of AL heuristics are used to identify instances of switching between the query strategies. This approach of implementing hybrid query strategies has not been presented in the literature before and is an important contribution of this work.

3.2 Active Learning Challenges

3.2.1 Cold-start problem in Active Learning

The initial labelled dataset is a key component in an AL framework. The expectation is that a small portion of the dataset is labelled and available to the learner for its use. However, acquiring labels for the initial labelled dataset can be extremely time-consuming, expensive, or infeasible in many practical situations [40]. This gives rise to the cold-start problem in AL settings, where there are no labels in the beginning to train an initial classification model. The proposed framework addresses this issue through a pre-clustering approach. In this approach, the unlabelled dataset is clustered using unsupervised clustering algorithm and the points closest to the centroids of each cluster are selected as the instances for an initial labelled dataset.

3.2.2 Oracle Uncertainty

The oracle is assumed to be infallible in AL settings [54]. In practical scenarios, the labels are obtained from experimental test set-ups or provided by a human expert. In either case, there is an uncertainty associated with the labels due to experimental errors, measurement noise or human mistakes [42]. This motivates the issue of oracle uncertainty in conventional Active Learning methodologies. This issue is addressed by incorporating additional feedback during the query process. This feedback takes the form of (i) a grade on the expertise of the human oracle and (ii) a confidence level of the expert in providing such label. The feedback helps the Active Learner to quantify the reliability of annotations obtained from the oracle.

3.2.3 Hybrid query strategies

AL frameworks employ querying strategies to select instances. These strategies include Uncertainty Sampling (US) [55–57] and Query-by-Committee (QBC) [58–60]. Other query strategies such as Expected Error Reduction [61, 62], Variance Reduction [63, 64] and Density-weighted Methods (DWM) [55, 65, 66] are described in the literature. The selection of query strategy for a specific application depends on several factors such as computational cost, feasibility to implement and compatibility with base learning methods. AL methodologies usually implement a single query strategy throughout the entire querying process. However, this is not convenient in practical scenarios. The reasons include a query strategy that might be easy to implement but fail to deliver satisfactory performance, or another strategy might exhibit stellar performance but possess high computational cost. This motivates the use of hybrid query strategies, which allows to combine the benefits of multiple methods within a single framework. For instance, a hybrid combination of two appropriate query strategies may help to reduce the computational cost, while exhibiting at par or better performance than the individual query strategies.

3.2.4 Absence of ground truth

The performance evaluation of AL methodologies is generally performed by comparing the predicted labels against the ground truth. Hence, the corresponding results are reported in the form of classification accuracy. However, acquiring ground truth labels for all the instances in a given dataset is not straightforward in practical settings due to the large amount of work involved in the labelling process. Moreover, the labelling process may also result in experimental or human errors, which degrade the quality of the labels. Due to the aforementioned shortcomings, classification accuracy is not an appropriate indicator of the performance of the Active Learner. This work proposes the use of heuristics evaluated during the iterative querying process, as metrics to mimic classification accuracy. These heuristics are observed to exhibit correlation with classification accuracy, and represent the performance of the Active Learner.

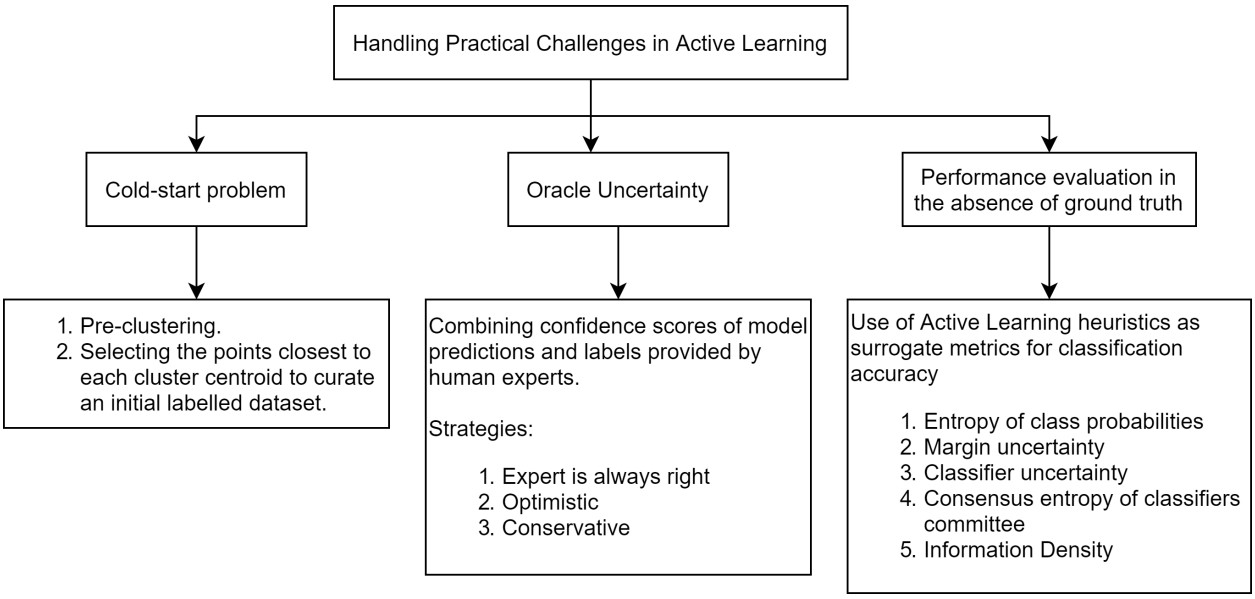


Figure 3.1: Proposed Active Learning framework for handling practical challenges

3.3 Proposed Framework

The proposed Active Learning framework incorporates a hybrid query strategy that addresses the following practical challenges: (i) cold-start problem, (ii) oracle uncertainty, (iii) use of hybrid query strategies, and (iv) performance quantification in the absence of ground truth labels. The proposed framework is presented in Figure 3.1. The cold-start problem is addressed using pre-clustering to assign labels to instances close to the cluster centroids. Meanwhile, the issue of oracle uncertainty in handled by making a combined decision based on the confidence scores of model predictions and those of the human experts. In addition, a set of metrics derived from Active Learning heuristics is proposed to evaluate the quality of the labels. These metrics are also used to identify the instance that requires switching to a different query algorithm under the proposed hybrid query strategy. A flowchart from implementing the proposed framework is demonstrated in Figure 3.2 and all the corresponding elements are elaborated as follows.

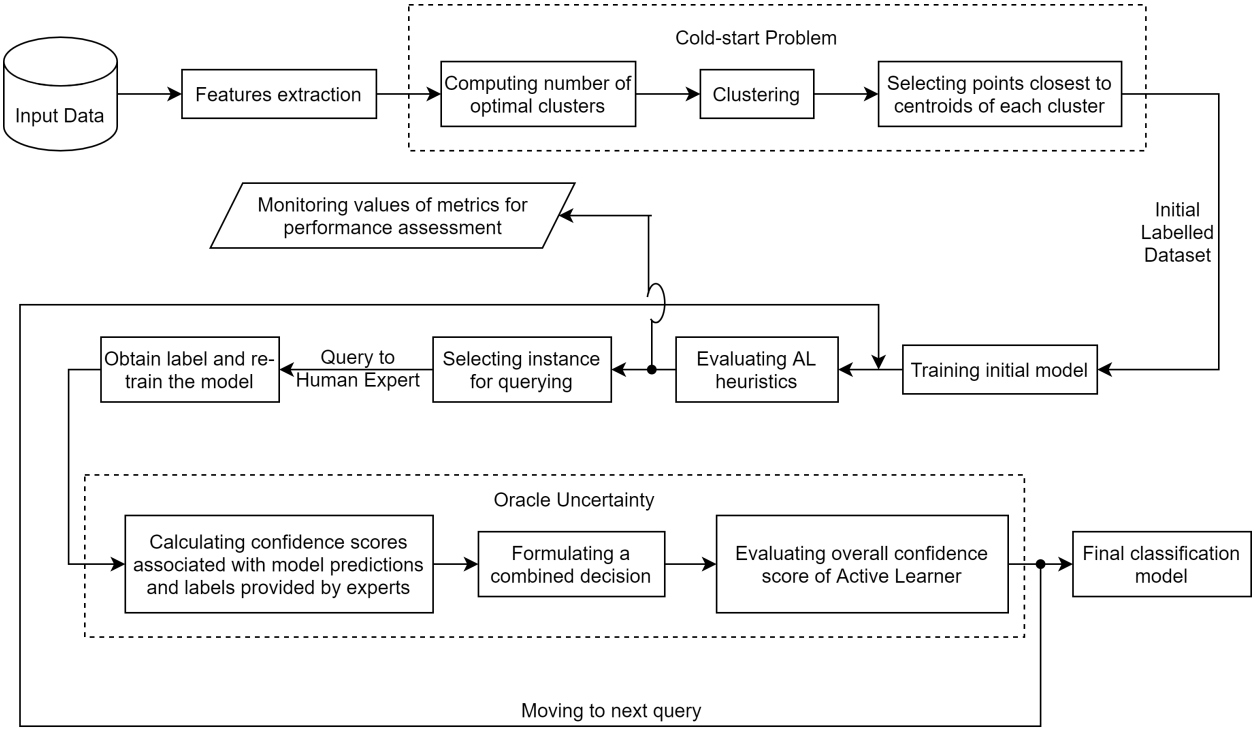


Figure 3.2: Flowchart for implementing the proposed AL framework to address practical challenges

3.3.1 Addressing cold-start problem

The proposed framework handles the cold-start problem in AL by introducing a pre-clustering step that generates the initial labelled dataset. Given a dataset with no labels, the procedure is as follows:

1. Select an unsupervised clustering algorithm and apply it to the dataset with no labels. In this work, k-means clustering has been used for the pre-clustering step as it is relatively simple to implement and scales to large datasets easily.

2. Identify the number of optimal clusters for the given dataset. The elbow method is employed to select the optimal number of clusters in this work, as discussed in [67].

3. Create the initial labelled dataset by selecting the points closest to the centroids of each cluster, such that the number of instances in the generated set do not exceed a pre-defined fraction of that in the complete dataset. This work considers the threshold to be 2%, i.e., the size of the initial labelled dataset should be 2% of the entire dataset.

Once the initial labelled dataset is obtained using the aforementioned procedure, it can be used to train the classification model, which is updated continuously during the iterative querying process in the AL framework.

3.3.2 Handling oracle uncertainty

The oracle uncertainty challenge originates in the ambiguity surrounding the expertise of the labeler and the confidence in the given labels. Thus, a solution to this challenge requires a way to incorporate this uncertainty via a confidence score that accounts for the model predictions and the input labels. The AL process produces a posterior probability associated to each class for all the instances. The posterior probabilities indicate the likelihood of a particular instance being classified into the given class per the current model. The posterior probability can be used to calculate heuristics such as the level of uncertainty, entropy of class probabilities and classification margin. The maximum value of the posterior probability is calculated using

$$p_{max} = \max\left(P_\theta(\hat{y}|\boldsymbol{x})\right), \tag{3.1}$$

where $\hat{y}$ represents the predicted label for the instance $\boldsymbol{x}$ under the model θ. Further, $p_{max} \in [0, 1]$ is scaled-up to the range of 1-5 to associate confidence scores with the model predictions. Here, 1 represents the lowest confidence score and 5 represents the highest confidence score.

In addition to the confidence scores of the model, the confidence scores supplied by the oracle need to be incorporated as part of this framework. The confidence score associated with labels supplied by the human experts is a result of the following inputs during the continuous querying process: (i) label, (ii) grade (level of expertise) of the human annotator (z_1) and (iii) confidence level corresponding to the label provided (z_2). z_1 and z_2 are numeric values between 1-5, where one represents the lowest expertise/confidence level and five represents the highest expertise/confidence level. Variables z_1 and z_2 are used to produce an overall confidence score of the expert on a specific label, as per the rule base shown in

Table 3.1: Rule to generate overall confidence score of the expert

Confidence Level (z_2)	**Grade of Expert (z_1)**				
	1	**2**	**3**	**4**	**5**
Not Provided	1	1	2	2	3
1	1	1	1	1	1
2	1	1	1	2	2
3	1	1	2	3	3
4	1	2	2	4	4
5	1	2	3	4	5

Table 3.1. It should be noted that the lower graded experts are penalized by imposing a lower confidence score than what is originally reported (z_2). This is done because there is relatively lesser trust on the lower graded experts. On the other hand, the scores provided by higher graded experts are accepted without any penalization.

The next step involves formulation of a combined decision based on the confidence scores of model predictions and the labels provided by human experts. This is implemented using three different strategies discussed as follows:

1. **Expert is always right:** The confidence score corresponding to the label provided by the expert is given a higher priority and is used to update the confidence score of the model during each query.

2. **Optimistic approach:** It considers the higher confidence score among that of the model and the expert.

3. **Conservative approach:** It considers the lower confidence score among that of the model and the expert.

Finally, the confidence scores of model and the expert are used to generate an overall confidence score of the Active Learner at each query step. It is calculated as the weighted average of the scores corresponding to all the instances in the dataset, as given by

$$S_{AL} = \frac{\sum_{i=1}^{5} s_i n_i}{\sum_{i=1}^{5} n_i}, \tag{3.2}$$

where, n_i represents the number of instances in the dataset with confidence score s_i.

3.3.3 Performance evaluation and hybrid query strategies

The fundamental premise of AL methodologies is the iterative selection of most informative instances by the optimization of selected heuristics. The framework here described proposes the use of the selected heuristics as metrics to quantify the performance of the Active Learner. This approach is beneficial in many practical scenarios, where it is not feasible to obtain ground truth labels for all the instances in a dataset. The metrics used in this work are presented in Table 3.2. Entropy of the Class probabilities (EC), Margin Uncertainty (MU) and Classifier Uncertainty (CU) are associated with uncertainty measures in US query strategy. Consensus Entropy of the committee of classifiers (CE) is derived from disagreement measures in QBC. Information densities with Euclidean distance (IE) and Cosine similarity (IC) are linked to the similarity measures in DWM approach.

Given that these metrics are derived from the heuristics involved in Active Learning settings, they are expected to mimic classification accuracy so that the performance of Active Learner can be evaluated in the absence of ground truth. For instance, the value of classifier uncertainty decreases with the most informative instances being labelled at each query step. Consequently, the value of CU should decrease with an increase in classification accuracy as more instances are labelled iteratively. Similarly, the values of EC and CE are also expected to decrease and MU is expected to increase in relation to classification accuracy as more queries are made. The performance evaluation of Active Learning methodologies using such heuristics has not been reported in the relevant literature so far to the best of our knowledge and stands out as a novel contribution of this work.

In this work, the metrics CU, MU, EC and CE are used, which can be calculated as follows:

- CU can be calculated by

$$\text{CU} = 1 - P_\theta(\hat{y}|\boldsymbol{x}) \tag{3.3}$$

- MU can be written as

$$\text{MU} = P_\theta(\hat{y_1}|\boldsymbol{x}) - P_\theta(\hat{y_2}|\boldsymbol{x}) \tag{3.4}$$

where, $\hat{y_1}$ and $\hat{y_2}$ are the first and second most likely predictions under the model θ, respectively.

- EC can be evaluated as

$$\text{EC} = -\sum_{y} P_\theta(y|\boldsymbol{x}) \log(P_\theta(y|\boldsymbol{x})) \tag{3.5}$$

where y ranges over all possible labelling of x.

- CE can be expressed as

$$\text{CE} = -\sum_{y} P_\mathcal{C}(y|\boldsymbol{x}) \log(P_\mathcal{C}(y|\boldsymbol{x})) \tag{3.6}$$

where,

$$P_\mathcal{C}(y|\boldsymbol{x}) = \frac{1}{|\mathcal{C}|} \sum_{\theta \in \mathcal{C}} P_\theta(y|\boldsymbol{x}) \tag{3.7}$$

$\mathcal{C}$ is the committee of classifiers and $|\mathcal{C}|$ is the size of the committee.

- IE can be computed as

$$\text{IE} = \frac{1}{|\mathcal{U}|} \sum_{\boldsymbol{x'} \in \mathcal{U}} d(\boldsymbol{x}, \boldsymbol{x'}) \tag{3.8}$$

where $\mathcal{U}$ represents the pool of unlabelled instances, $|\mathcal{U}|$ is the size of the pool of unlabelled instances and $d(\boldsymbol{x}, \boldsymbol{x'})$ is the Euclidean distance between $\boldsymbol{x}$ and $\boldsymbol{x'}$, given by

$$d(\boldsymbol{x}, \boldsymbol{x'}) = \sqrt{(x_1 - x'_1)^2 + (x_2 - x'_2)^2 + ... + (x_n - x'_n)^2} \tag{3.9}$$

n is the size of feature vectors used for training the classification models.

- IC can be determined as

$$\text{IC} = \frac{1}{|\mathcal{U}|} \sum_{\boldsymbol{x'} \in \mathcal{U}} sim(\boldsymbol{x}, \boldsymbol{x'}) \tag{3.10}$$

Table 3.2: Details of the proposed metrics for the hybrid query strategy-based AL framework

Serial Number	Metric Description	Abbreviation	Remarks
1	Entropy of the class probabilities	EC	Measures of uncertainty
2	Margin uncertainty, i.e., difference of the probabilities of first and second most likely predictions	MU	
3	Classifier uncertainty, i.e., 1 - P(correct prediction)	CU	
4	Consensus entropy of the committee of classifiers in QBC approach	CE	Measure of disagreement
5	Information density with Euclidean distance as similarity measure	IE	Density-weighted heuristics
6	Information density with cosine similarity as similarity measure	IC	

where $sim(\boldsymbol{x}, \boldsymbol{x'})$ is the cosine similarity between $\boldsymbol{x}$ and $\boldsymbol{x'}$, given by

$$sim(\boldsymbol{x}, \boldsymbol{x'}) = \frac{\boldsymbol{x} \cdot \boldsymbol{x'}}{||\boldsymbol{x}||\,||\boldsymbol{x'}||} \tag{3.11}$$

The values of the metrics enlisted in Table 3.2 can also be examined to identify suitable instances of switching to a different query strategy rather than using a same strategy throughout the querying process. This approach is highly advantageous in numerous industrial settings, where it is desirable to have a trade-off between aspects such as labelling expenditures, computational cost and appreciable performance of the Active Learner. The query strategy can be switched when the values of metrics either oscillate beyond pre-defined thresholds or do not exhibit a substantial change over a certain number of queries. Moreover, the decision regarding termination of querying process can also be made when the metrics values exhibit saturation or approach a pre-determined threshold. The use of hybrid strategies is motivated by the equal or improved performance when compared to individual query strategies at a lower computational cost.

3.4 Experimental Evaluation

In order to evaluate the robustness of our proposed Active Learning framework across different environments and industrial settings, the proposed framework is implemented in the

following datasets: (i) process data collected from a simulated ethanol plant [68], (ii) drug consumption (quantified) dataset from UCI Machine Learning repository [69], and (iii) rolling element bearing test data from the bearing data center at Case Western Reserve University [70]. The details of the analyses and results are described below.

3.4.1 Description of the datasets

Dataset 1 - Process data collected from a simulated ethanol plant:

The data from a simulated ethanol plant is used to describe how to overcome the clod-start challenge. The simulations are based on mass and energy balances and the corresponding Human Machine Interface (HMI) is designed using MATLAB. Details of the simulated ethanol plant are presented in [68]. It consists of a continuous stirred tank reactor (CSTR) and a distillation column (DC). The measurements available correspond to 11 process variables: ethanol concentration in CSTR (C101), feed flow rate to CSTR (F101) and DT (F105), coolant flow rate (F102), level of CSTR (L101), temperatures of coolant inlet (T101), coolant outlet (T102), CSTR (T103), third tray of DT (T104), fifth tray of DT (T105) and eighth tray of DT (T106) [68]. There are 4 different scenarios within these measurements: normal operating conditions (14041 instances), coolant-related disturbances (1647), CSTR-related disturbances (2819) and reflux imbalance (9336), thereby constituting a total of 27843 instances. The output of these 11 processes are treated as features to the classification problem. The output data originates from an experiment where 10 participants were asked to control the process variables manually if their values extend beyond the normal operating range.

Dataset 2 - Drug consumption (quantified) dataset:

This dataset contains details of 1885 participants regarding their usage of various types of legal and illegal drugs. The features for this dataset comprise of personality measurements, which are termed as NEO-FFI-R (neuroticism, extraversion, openness to experience, agreeableness, and conscientiousness), BIS-11 (impulsivity), and ImpSS (sensation seeking). In addition, it includes level of education, age, gender, country of residence and ethnicity as

features too [69]. The categorical input attributes are quantified and used as numerical values [69]. The labels contain information about usage of cannabis drugs and consist of seven classes: "Never Used", "Used over a Decade Ago", "Used in Last Decade", "Used in Last Year", "Used in Last Month", "Used in Last Week", and "Used in Last Day". The classes are reorganized to form a "year-based" binary classification problem as described in [69]: the classes "Never Used", "Used over a Decade Ago" and "Used in Last Decade" are merged to form a group of cannabis non-users and all other classes are combined to form a group of cannabis users.

Dataset 3 - Rolling element bearing test data:

This dataset consists of ball bearing test data obtained from experiments conducted using a 2 hp electric motor under normal and faulty conditions. The faults vary in size (0.007" to 0.040" in diameter) as well as location (inner raceway, outer raceway, or rolling element). The vibration data of the motor captured at the drive end for loads in the range 0-3 hp are used. This corresponds to four different motor speeds between 1720-1797 RPM. The data is sampled at 12 kHz for 10 seconds at each speed level. In this experiment, each of these signal segments are split into 40 instances of 0.25 seconds each at each speed level. It results in 160 (40 x 4 speed levels) instances for each normal and fault condition. Furthermore, the classification problem is formulated by selecting the fault size as 0.021" and having four classes corresponding to the fault location - healthy, inner-race, ball and outer-race. Consequently, there are a total of 640 (160 x 4 classes) instances used in this experiment. Wavelet Packet Decomposition (WPD) technique is employed for extracting features from these signals, as discussed in [71]. WPD is performed at level 6 using mother wavelet from Daubechies family as it has been reported to be used in other health management problems [71]. Furthermore, the wavelet coefficients that correspond to different frequency bands are extracted and their Shannon entropy values are used as features for classification problem. WPD analysis is performed using the Wavelet toolbox in MATLAB.

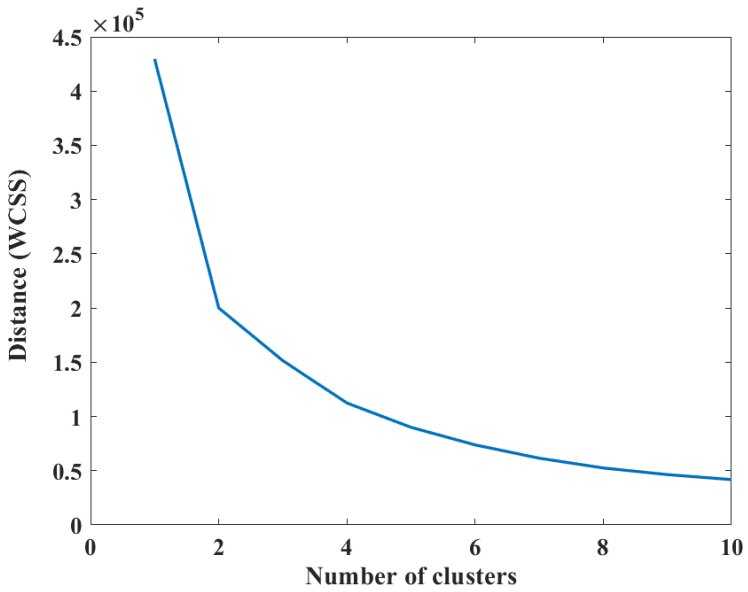


Figure 3.3: Plot of WCSS vs. number of clusters - Dataset 2: Drug Consumption (quantified) dataset.

3.4.2 Experimental Settings

Dataset 1 - Process data collected from a simulated ethanol plant:

The initial labelled dataset is pre-processed with the method described in Section 3.3.1. The optimal number of clusters is identified as four using the elbow method. The plot of within-cluster sum of squares distance (WCSS) vs. number of clusters is shown in Figure 3.3. It can be observed that after four clusters, the rate of change in the WCSS distance decreases significantly. An "elbow" is visible at four-clusters point. The size of the initial labelled dataset is set as 500 instances with 125 instances for each class.

Once the initial labelled dataset is formed, the first model is trained and the iterative querying process begins. QBC is implemented in Python using the modAL framework [72] as the basic tool for selection of instances for annotation by a human expert. k-Nearest Neighbors (kNN) is selected as the base classification method because it is simple and easy to implement, versatile, and is a non-parametric algorithm. Moreover, it does not make any assumptions on the input data distribution. The metrics introduced in Section 3.3.3 are also evaluated at each query step along with the classification accuracy.

Dataset 2 - Drug consumption (quantified) dataset:

This is a binary classification problem, where the given participant may be a cannabis user or non-user. The initial labelled dataset is created by selecting 2% of the total instances at random along with corresponding labels. An initial model is formed by using the initial labelled dataset and then the iterative querying process is initiated. In these experiments, US and QBC are implemented for querying in Python using the modAL framework [72] and kNN is chosen as the base classifier. In the experiment using this dataset, three query procedures are used. These procedures are: (i) all queries using US, (ii) all queries using QBC, and (iii) hybrid approach, i.e., first half of the queries using US and another half using QBC. All the relevant metrics are recorded and monitored at each query step for performance evaluation in the absence of ground truth and selection of instances for switching between the query strategies.

Dataset 3 - Rolling element bearing test data:

The dataset is pre-processed using WPD to obtain the features for all the instances. The procedure described in Section 3.3.1 is applied to the resulting instances to overcome the cold-start problem. Using the elbow method, the optimal number of clusters is set to four. Once the initial labelled dataset is available, the first model is trained and the iterative querying process is initiated. The modAL framework [72] is used for implementation of US and QBC query strategies.

A total of 640 normally distributed random numbers in the range of 1-5 are used to generate variables z_1 and z_2. The normally distributed random numbers help to simulate the difficulty of finding many maintenance and vibration experts with different levels of skills. The resulting overall confidence score of the expert is produced using variables z_1 and z_2 according to the procedure described in Section 3.3.2. Afterwards, the overall confidence score of the Active Learner is calculated using Eq. 3.2 for all the three strategies enlisted in Section 3.3.2. Finally, the relevant metrics are calculated and monitored at each query step for different purposes, as highlighted in Section 3.3.3.

Table 3.3: Results for proposed metrics - Dataset 1: Process data collected from a simulated ethanol plant

Metrics	Unqueried Value	Values after making no. of queries				
		500	1000	1500	2000	2500
EC	0.38	0.35	0.29	0.26	0.23	0.21
MU	0.81	0.83	0.84	0.86	0.87	0.9
CU	0.14	0.12	0.11	0.1	0.08	0.07
CE	0.18	0.15	0.11	0.09	0.06	0.04
IE	0.05	0.05	0.06	0.06	0.06	0.07
IC	0.96	0.97	0.97	0.97	0.98	0.98
Classification Accuracy	0.71	0.77	0.81	0.87	0.91	0.97
Classification Accuracy - using randomly generated initial labelled dataset	0.69	0.72	0.77	0.84	0.89	0.96

3.4.3 Results

The advantages of implementing a pre-clustering approach to alleviate the cold-start problem is illustrated using experiments performed on ethanol plant dataset. Table 3.3 presents the results of the Active Learning framework within the context of the cold-start challenge with the pre-clustering step incorporated. The entries in the column *Unqueried value* represent the metrics values when the model is trained with only the initial labelled dataset and before any queries were carried out. The entries in subsequent columns indicate the metrics values corresponding to the updated model after reaching the indicated number of queries. Firstly, it can be observed that the classification accuracy increases as more queries are made. This indicates that the performance of the Active Learner improves during the iterative querying process. Moreover, it is worthwhile to note that classification accuracy reaches 91% with 2000 queries, which is just around 7% of the total instances in the dataset.

Table 3.3 shows that EC, MU, CU and CE exhibit a correlation with the classification accuracy. EC, CU and CE are observed to decrease with increased classification accuracy as more instances are queried. This is expected because entropy of the class probabilities, level of uncertainty and level of disagreement between the committee members (QBC approach) decrease as the most informative instances are labelled by the Active Learner during the initial set of iterations. On the other hand, MU grows with increased classification accu-

Table 3.4: Results for hybrid query strategy - Dataset 2: Drug consumption (quantified) dataset

Metrics	Unqueried value		Values after making no. of queries									
			125		250		375			500		
	US	QBC	US	QBC	US	QBC	US	QBC	*US + QBC*	US	QBC	*US + QBC*
EC	0.64	0.58	0.56	0.51	0.52	0.48	0.47	0.44	*0.46*	0.43	0.41	*0.41*
MU	0.27	0.38	0.34	0.44	0.42	0.49	0.48	0.56	*0.5*	0.52	0.63	*0.57*
CU	0.36	0.32	0.33	0.29	0.31	0.24	0.27	0.21	*0.26*	0.25	0.19	*0.23*
CE	0.53	0.48	0.46	0.42	0.43	0.39	0.39	0.36	*0.38*	0.35	0.34	*0.34*
IE	0.21	0.21	0.21	0.22	0.22	0.22	0.22	0.23	*0.23*	0.23	0.23	*0.23*
IC	0.52	0.51	0.52	0.51	0.52	0.51	0.51	0.51	*0.51*	0.51	0.5	*0.51*
Classification Accuracy	0.58	0.69	0.64	0.74	0.76	0.8	0.85	0.88	*0.89*	0.89	0.9	*0.91*

racy as more instances are queried. The reason is a higher margin implies less uncertainty in distinguishing between the two most likely alternatives and is likely to increase with increasing number of queries. The metrics IE and IC do not exhibit an appreciable change as more instances are queried because information density is not considered while selecting instances using QBC query strategy. These heuristics might change when using Density Weighted Methods for instance selection during querying. Hence, the proposed framework can be used to evaluate the performance of Active Learner using the metrics in the absence of ground truth information. Secondly, the classification accuracies corresponding to the proposed framework is a bit higher as compared to the case when the initial labelled dataset would have been obtained by means of random sampling. This is an added advantage of the pre-clustering step in the proposed Active Learning framework.

The benefits of hybrid query strategies are established using experiments performed on the drug consumption dataset and the corresponding results are shown in Table 3.4. Here, 500 queries (i.e., around 25% of total instances in the dataset) are made using three different modes: (i) all queries using US, (ii) all queries using QBC, and (iii) a hybrid of US and QBC - first half of the queries using US and next half using QBC. In all the querying procedures, the classification accuracy increases as more queries are made, thereby indicating the improving performance of the Active Learner during the querying process.

It can be seen from the values of classification accuracy that the performance of the Active Learner in the hybrid case is at par or better than that of the individual query strategies.

The US strategy has greater ease of implementation when compared to QBC but it provides lower performance results than QBC. On the other hand, QBC is computationally expensive when compared to US, although it achieves a superior performance in terms of classification accuracy. The hybrid approach of US + QBC helps to accomplish a trade-off between these two aspects - the performance at the end of 500 queries is better than that of individual query strategies, while simultaneously reducing the expense of QBC as first 250 queries are made using US, which is comparatively cheaper in terms of computational cost. Finally, all the metrics for the three querying procedures exhibit a correlation with classification accuracy as shown in Table 3.4. This correlation is an expected behavior given the relationship between the metrics and classification accuracy.

The contribution of all the elements together in the proposed framework is exhibited through experiments performed on rolling element bearing dataset. The results of this experiment that aims to evaluate the performance of an hybrid query strategy are shown in Table 3.5, where the hybrid combination of US + QBC is observed to deliver at par or better performance than individual query strategies. The trade-off between model quality and computational cost provides an additional incentive for the use of the hybrid strategy. When starting with a dataset without labels, the classification accuracy, relevant metrics and the overall confidence scores of the Active Learner using three combination strategies are reported in Table 3.6. EC, CU and CE exhibit an inversely proportional correlation with classification accuracy and MU is directly proportional to classification accuracy as more queries are made. This behavior of the metrics is consistent with the previous dataset experiments.

Furthermore, it can be observed that the overall confidence scores of the Active Learner obtained by combining those of the model predictions and the experts exhibit an increasing trend with increasing number of queries for all the strategies. The plot showing this performance over 30-52 queries is presented in Figure 3.4. This range of queries is selected for better readability and comparing performance across different combination strategies. The intermediate valleys in the plot are probably due to outliers being selected by the Active Learner for annotation by the human experts. It can be observed that the overall confidence

Table 3.5: Results for hybrid query strategy - Dataset 3: Rolling element bearing test data from the bearing data center at Case Western Reserve University

Metrics	Unqueried value		Values after making number of queries									
			25		50		75			100		
	US	QBC	US	QBC	US	QBC	US	QBC	*US + QBC*	US	QBC	*US + QBC*
EC	0.93	0.72	0.77	0.64	0.62	0.52	0.53	0.44	*0.47*	0.48	0.38	*0.34*
MU	0.15	0.32	0.28	0.39	0.39	0.51	0.49	0.59	*0.51*	0.53	0.65	*0.59*
CU	0.51	0.37	0.45	0.32	0.41	0.25	0.37	0.23	*0.28*	0.27	0.19	*0.18*
CE	0.89	0.67	0.73	0.59	0.61	0.51	0.50	0.43	*0.43*	0.46	0.34	*0.30*
IE	0.01	0.01	0.02	0.02	0.02	0.01	0.02	0.02	*0.02*	0.03	0.02	*0.03*
IC	0.58	0.58	0.58	0.58	0.59	0.58	0.59	0.59	*0.59*	0.60	0.59	*0.60*
Classification Accuracy	0.40	0.53	0.54	0.67	0.69	0.86	0.74	0.93	*0.89*	0.88	0.98	*0.99*

Table 3.6: Results for proposed metrics - Dataset 3: Rolling element bearing test data from the bearing data center at Case Western Reserve University

Metrics	Unqueried Value	Values after making no. of queries				
		25	50	75	100	125
Confidence Score (Expert is always right)	3.02	3.6	3.93	4.45	4.46	4.70
Confidence Score (Optimistic approach)	3.03	3.64	3.94	4.46	4.46	4.71
Confidence Score (Conservative approach)	3.03	3.64	3.93	4.46	4.46	4.70
EC	0.93	0.80	0.66	0.57	0.42	0.38
MU	0.16	0.24	0.40	0.57	0.75	0.78
CU	0.47	0.41	0.38	0.31	0.25	0.16
CE	0.84	0.74	0.61	0.51	0.41	0.38
IE	0.01	0.01	0.02	0.02	0.02	0.03
IC	0.58	0.59	0.59	0.59	0.60	0.61
Classification Accuracy	0.45	0.59	0.74	0.79	0.91	0.98
Classification Accuracy - using randomly generated initial labelled dataset	0.40	0.54	0.69	0.74	0.88	0.98

scores corresponding to the optimistic strategy is generally higher than the other strategies. This is intuitive as the optimistic approach considers higher confidence scores among that of the model and the expert.

3.5 Summary

This chapter introduces a hybrid query strategy-based framework, which addresses the practical challenges in AL, namely, cold-start, oracle uncertainty and evaluating model performance in the absence of ground truth. The robustness of the proposed framework is evaluated

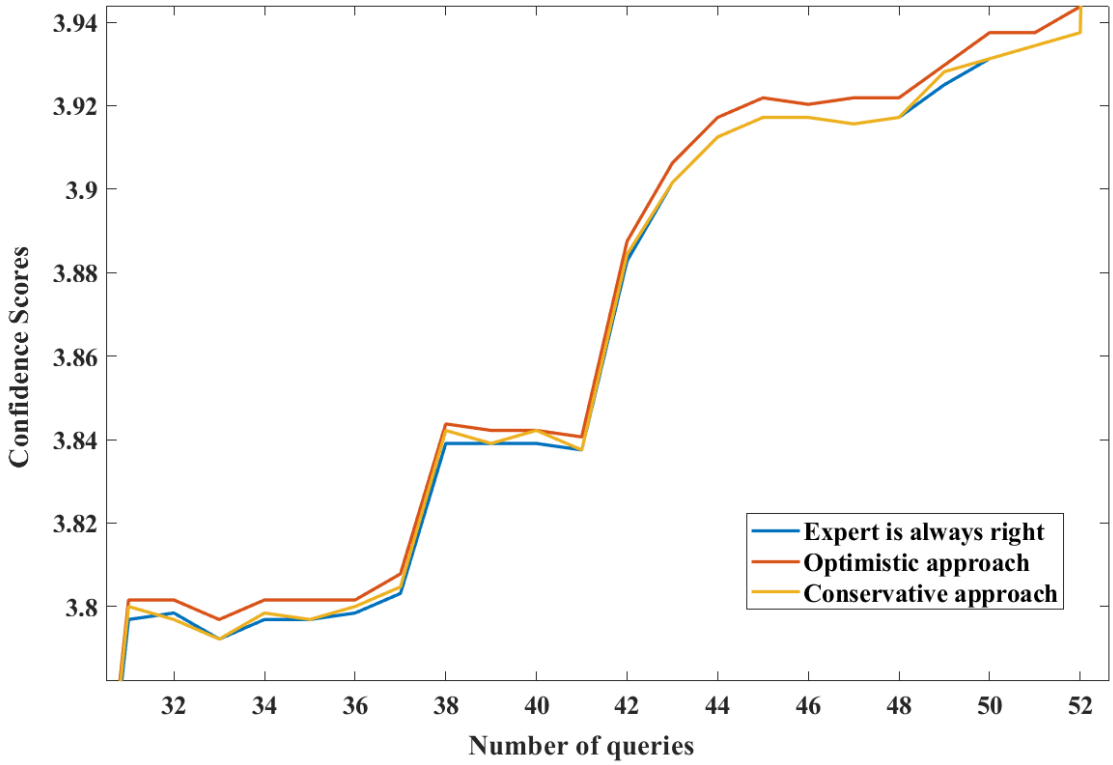


Figure 3.4: Plot of overall model confidence scores with number of queries made to the expert - Dataset 3: Rolling element bearing test data from the bearing data center at Case Western Reserve University.

across three datasets from different environments and industrial settings. Firstly, the merits of using a pre-clustering approach to address the cold-start problem is exemplified by experiments performed on the process data collected from a simulated ethanol plant. Secondly, the utility of hybrid query strategies is established with the help of experiments performed on the drug consumption dataset. It is concluded that the use of hybrid query strategies helps in reducing the computational cost, while delivering at par or better performance than the individual query strategies. Finally, the contribution of all the elements together in the proposed framework is demonstrated through experiments carried out on the rolling element bearing test data from Case Western Reserve University. In the next chapter, the idea of AL shall be extended to representation learning in non-Euclidean space like graphs.

Chapter 4

Active Learning over Graphs

In recent years, there has been a substantial development in AI engines and machine learning technologies to harness the power of data and build efficient decision models that are capable of making accurate predictions and inferences. However, a huge amount of labeled data is required to train these computational models. Semi-supervised learning approaches like AL helps to address this limitation by actively and strategically choosing the most informative samples from the pool of unlabeled data. Consequently, it is possible to accomplish a better predictive performance with a significantly reduced training sample size.

There is an increasing research interest in analyzing and learning from data in non-Euclidean space, like graphs. It has a wide variety of applications in several interesting areas like social networks, knowledge graphs, protein-protein interaction networks, combinatorial optimization and molecular fingerprinting. Graph Neural Networks (GNN) provide an elegant framework for effective inferencing on such graph-structured data. Given the extensive applications of learning from graph-structured data and the benefits of semi-supervised learning techniques, it would be highly beneficial to consolidate AL with GNN for many downstream tasks like node classification, link prediction, community detection and graph classification. This chapter presents an AL framework combined with GNN for node classification in attributed graphs using a convex optimization approach.

The node classification task in attributed graphs is addressed by utilizing a framework

that combines AL and GNN using a rigorous convex optimization approach. The novel contributions are as follows:

- Four graph-theoretic metrics are used as AL heuristics/acquisition functions for active selection of nodes.
- Both node attributes and topological information are incorporated in the learning scheme. The node features are exploited while training the GNN-based decision model and topological information is considered during selective sampling of the nodes.
- An inductive GNN approach is used for building node classification models. This allows the approach to be scalable across graphs of different sizes as well as subgraphs within the given graph.

4.1 Related Work

There is only a limited number of AL models for graph-structured data as compared to other types of data that are represented in Euclidean space, like images, text or numerical data [73]. It has also been established that the use of graph theoretic measures as acquisition functions in AL provide better performance than using traditional heuristics like classifier uncertainty, entropy of class probabilities or classification margin [74]. The early works related to application of AL on graph data are based on cluster assumption and local and global consistency [75, 76]. In these techniques, it is assumed that nearby points on the same structure (i.e., cluster or manifold) are likely to have the same labels. This category of AL methodologies do not mimic realistic settings and restricts the modeling capacity.

Some of the works related to implementing AL of graph-structured data are linked to earlier classification models like Gaussian random fields [77–79]. However, these models do not incorporate node features and label information in the learning process. Moreover, some of these models are not adaptive in the sense that the active learner is not updated based on the newly labeled instances. These limitations are addressed by approaches that employ AL models coupled with GNN architectures [74, 80]. However, such studies are very limited

and the corresponding acquisition functions are based only on matrix concepts for centrality formulation. This results in a lack of intuition related to the underlying network topology.

In this chapter, a convex optimization driven AL framework combined with GNN is presented, that uses four graph theoretic measures, namely, eigenvector centrality, closeness centrality, betweenness centrality and Effective Graph Resistance (EGR) as AL heuristics for active node selection in attributed graphs.

4.2 Convexity Analysis

The following result formalises the convexity of the optimization problem for Active Learning.

Theorem 1. *The optimization problem for Active Learning, given by:*

$$min\, \lambda ||\boldsymbol{SZ}||_{q,1} + tr(\boldsymbol{D}^T \boldsymbol{Z})\ s.t.\ \boldsymbol{Z} \geq 0, \mathbf{1}^T \boldsymbol{Z} = \mathbf{1}^T \tag{4.1}$$

over the optimization variable $\boldsymbol{Z} \in \mathbb{R}^{|\mathcal{U}| \times |\mathcal{U}|}$ *is convex, where* $\boldsymbol{D}$ *is the dissimilarity matrix indicating the degree of dissimilarity between samples (nodes) in the dataset (graph),* $\boldsymbol{S}$ *is the overall score matrix formed by combining informativeness and diversity scores of all the samples in the dataset and* λ *is a parameter that balances the costs associated with encoding all the samples and number of representatives.*

Proof. It is well known that a function $f : \mathbb{R}^n \to \mathbb{R}$ is convex if $\text{dom}(f)$ is a convex set and if for all x_1, $x_2 \in \text{dom}(f)$, and $0 \leq \theta \leq 1$, we have

$$f(\theta x_1 + (1-\theta)x_2) \leq \theta f(x_1) + (1-\theta) f(x_2) \tag{4.2}$$

Consider $||\boldsymbol{SZ}||$

$$||\boldsymbol{S}(\theta \boldsymbol{A}_1 + (1-\theta)\boldsymbol{A}_2)|| = ||\theta \boldsymbol{SA}_1 + (1-\theta)\boldsymbol{SA}_2|| \tag{4.3}$$

By triangle inequality for matrix norms,

$$\begin{aligned} ||\boldsymbol{S}(\theta\boldsymbol{A}_1 + (1-\theta)\boldsymbol{A}_2)|| &\leq ||\theta\boldsymbol{S}\boldsymbol{A}_1|| + ||(1-\theta)\boldsymbol{S}\boldsymbol{A}_2|| \\ &= \theta||\boldsymbol{S}\boldsymbol{A}_1|| + (1-\theta)||\boldsymbol{S}\boldsymbol{A}_2|| \end{aligned} \tag{4.4}$$

This indicates that $||\boldsymbol{S}\boldsymbol{Z}||$ is convex. Further, λ is a positive constant multiplier. Hence, $\lambda||\boldsymbol{S}\boldsymbol{Z}||$ is also convex.

Consider the second term, $\text{tr}(\boldsymbol{D}^T\boldsymbol{Z}) = \sum_{i,j} d_{ij}z_{ij}$.

$$\begin{aligned} \text{tr}(\boldsymbol{D}^T(\theta\boldsymbol{A}_1 + (1-\theta)\boldsymbol{A}_2)) &= \sum_{i,j} d_{ji}(\theta a_1 + (1-\theta)a_2)_{ji} \\ &\leq \sum_{i,j} \theta d_{ji}a_{1ji} + \sum_{i,j}(1-\theta)d_{ji}a_{2ji} \\ &= \theta\sum_{i,j} d_{ji}a_{1ji} + (1-\theta)\sum_{i,j} d_{ji}a_{2ji} \\ &= \theta\,\text{tr}(\boldsymbol{D}^T\boldsymbol{A}_1) + (1-\theta)\,\text{tr}(\boldsymbol{D}^T\boldsymbol{A}_2) \end{aligned} \tag{4.5}$$

This indicates that $\text{tr}(\boldsymbol{D}^T\boldsymbol{Z})$ is convex. Since both the terms of eq. (4.1) are convex, their sum is also convex. The linear equality constraints ensures that the overall formulation in eq. (4.1) is convex. □

4.3 Methodology

This chapter proposes an AL framework for active node classification on attributed graphs. The proposed framework is shown in Figure 4.1. The raw input graph-structured data is preprocessed so as to convert it into a format suitable for further operations and analysis. After appropriate preprocessing steps, training, validation and test masks are generated to split the original graph-structured dataset into respective components. This is followed by definition of GNN model architecture (no. of GNN layers, no. of feed-forward layers, hidden dimensions, dropout, optimizer, epochs, etc.) and training an initial model using the initial

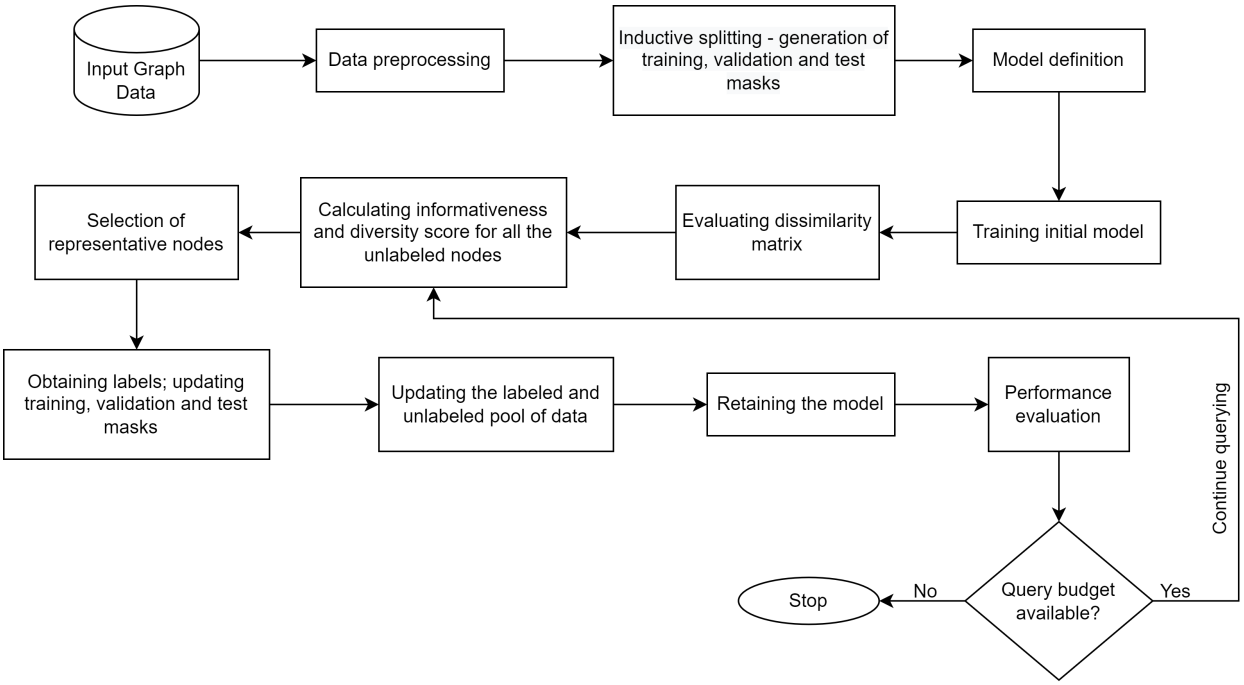


Figure 4.1: Proposed AL framework for active node classification on attributed graphs.

labeled dataset. The dissimilarity matrix $\boldsymbol{D} = \{d_{ij}\}_{i,j=1,2,3,\ldots,N}$ is evaluated by computing normalized Euclidean distances between node embeddings of all node pairs i, j.

The selection of representative nodes is initiated by calculation of informativeness ($s_{inf}(u)$) and diversity ($s_{div}(u)$) scores for all the nodes in the unlabeled pool. $s_{inf}(u)$ is obtained based on the values of AL heuristics. This step is similar to computing heuristics like classifier uncertainty, classification margin or entropy of class probabilities in conventional iterative querying AL frameworks [53, 81]. In this work, the following graph theoretic measures are used as acquisition functions to select the representative nodes in the graph-structured data.

1. Eigenvector centrality

   The most informative node from the unlabeled pool of nodes is given as:

$$u^* = \arg\max_u \frac{1}{\kappa} \sum_{j=1}^{N} A_{uj} c_j \tag{4.6}$$

   Here, c_j is the centrality of node j, A_{uj} ensures that only the neighbors of node u contribute to the sum and κ is the largest eigenvalue of A.

This metric tries to select the node which is connected to many other nodes in the network, while simultaneously weighing all the neighbors based on their importance or influence. It overcomes the limitation of degree centrality measure (already used in the literature [73]), which considers equal weights for all the neighbors of the node.

2. Closeness centrality

   The most informative node is selected using:

$$u^* = \arg\max_u \frac{N}{\sum_{j=1}^{N} d_{uj}} \tag{4.7}$$

   Here, d_{uj} is the shortest distance between the nodes u and j.

   The closeness centrality metric tries to select the node whose average distance to other nodes in the graph is smaller.

3. Betweenness centrality

   The most representative node is expressed as:

$$u^* = \arg\max_u \sum_{s=1}^{N} \sum_{t=1}^{N} n_{st}^u \tag{4.8}$$

$$n_{st}^u = \begin{cases} 1, \text{if } u \text{ lies on the shortest path from } s \text{ to } t \\ 0, \text{otherwise} \end{cases} \tag{4.9}$$

   This measure selects the nodes which have maximum control over information passing between other nodes in the graph.

4. Effective Graph Resistance (EGR)

   The most informative node from the unlabeled pool of nodes is selected using:

$$u^* = \arg\max_u (R_G - R_{Gu}) \tag{4.10}$$

Table 4.1: Summary of the datasets considered. Avg. Deg.: Average Degree, Avg. CC: Average Clustering Coefficient

Dataset	Nodes	Links	Features	Classes	Avg. Deg.	Avg. CC
Cora	2708	5429	1433	7	4.00	0.24
CiteSeer	3312	4732	3703	6	2.84	0.17

Here, R_G is the EGR of complete graph G and R_{Gu} represents the EGR of G after removal of node u.

This metric leads to selection of the nodes whose removal maximizes the decrease in graph robustness.

The aforementioned graph theoretic measures have not been used as AL heuristics/acquisition functions in the literature of node classification using active learning, and is a novel contribution of this work. The advantages of the proposed AL framework are multifold. Firstly, it incorporates both node attributes and topological information in the learning scheme. The node features are used while training the GNN-based decision model and topological information is considered during selective sampling of the nodes. Unlike a majority of existing works in the literature, inductive GraphSAGE [17] approach is used for building node classification models. This allows the approach to be scalable across graphs of different sizes as well as subgraphs within the given graph.

4.4 Experimental Evaluation

4.4.1 Datasets Description

The proposed AL framework is evaluated by performing experiments over two datasets, namely, Cora [82] and CiteSeer [83]. These are the citation networks consisting of scientific publications. Every node in the graphs represents a publication characterized by a set of binary features indicating the presence/absence of unique words from the dictionary. The edges in the graphs signify citation links among different publications. The properties of the datasets are summarized in Table 4.1.

4.4.2 Experimental Settings

All the downstream tasks related to representing and processing graphical data are handled using NetworkX [84] library in Python. PyTorch [85] is used to implement aspects of deep learning and GNN-specific tasks are integrated using Deep Graph Library (DGL) [86]. The detailed architecture of the GNN is as follows: depth (i.e., number of node embedding modules): 2; number of neurons in 2 layers: 48, 48; number of multi-layer perceptron (MLP) layers: 2; activation function: ReLU (except last layer with softmax); aggregation function: mean.

The training of GNNs is carried out in a mini-batch manner. 1.5 - 2% of the total nodes in the graph-structured datasets are randomly chosen to construct the initial labeled dataset. Around 20 - 25% of the nodes (uniformly selected over all classes) are kept aside for testing and are not used during training and validation steps. A total of 100 queries are made on Cora and 120 queries ($\sim$ 3.5 - 4% of the total nodes) on CiteSeer datasets. AL models are trained by considering four different graph-theoretic measures, namely, eigenvector centrality, closeness centrality, betweenness centrality and EGR (described in Section 4.3) as heuristics for selecting the most informative nodes. The evaluation is repeated for 100 times and the average values of classification accuracy scores are reported in Section 4.4.3. The degree centrality and clustering coefficient (CC) metrics, used in [73] are treated as the baselines. Degree centrality of a node u represents the fraction of nodes in the networks that are connected to u. The use of this metric as an AL heuristic results in selection of a node that is connected to maximum number of nodes in the network. On the contrary, CC of a node u is evaluated by computing the fraction of triangles possible through it. When used an acquisition fuction in AL framework, it tries to select nodes that possess high tendency to form clusters together.

4.4.3 Results and Discussion

The values of classification accuracy using all the four graph-theoretic measures (eigenvector centrality, closeness centrality, betweenness centrality and EGR) as well as the baselines

Table 4.2: Results of proposed AL methodology for Cora dataset

Query Strategy	**Initial Classification Accuracy**	**Classification Accuracy after no. of queries**			
		25	50	75	100
Degree	29.43%	46.42%	58.57%	68.28%	75.14%
Clustering Coefficient	27.66%	43.51%	52.17%	65.96%	69.86%
Eigenvector	32.64%	45.79%	64.43%	75.84%	77.92%
Closeness	29.71%	42.06%	50.28%	54.42%	60.97%
Betweenness	*30.42%*	*54.71%*	*67.85%*	*70%*	*80.57%*
EGR	*31.75%*	*52.36%*	*68.79%*	*75.53%*	*83.81%*

Table 4.3: Results of proposed AL methodology for CiteSeer dataset

Query Strategy	**Initial Classification Accuracy**	**Classification Accuracy after no. of queries**			
		25	50	75	100
Degree	27.64%	34.06%	46.18%	60.69%	71.56%
Clustering Coefficient	24.33%	37.57%	40.82%	58.48%	63.12%
Eigenvector	28.14%	39.51%	50.42%	67.31%	75.73%
Closeness	26.53%	35.56%	44.10%	62.54%	72.02%
Betweenness	*30.27%*	*41.78%*	*52.37%*	*63.22%*	*77.82%*
EGR	*30.91%*	*43.88%*	*56.25%*	*68.48%*	*79.38%*

(degree centrality and clustering coefficient) for Cora and CiteSeer datasets are tabulated in Table 4.2 and Table 4.3 respectively. The corresponding plots are shown in Figure 4.2 and Figure 4.3 respectively. It can be observed that two of the metrics, i.e., betweenness centrality and EGR consistently exhibit better performance as compared to both the baselines for Cora and CiteSeer datasets.

Degree centrality is a naive graph theoretic measure which quantifies the number of connections of a node to other nodes in the graph. On the other hand, the metrics like betweenness centrality and EGR incorporate topological information in a more elegant way as compared to degree centrality. In the context of AL framework, betweenness centrality tries to select the node having maximum control over information passing between other nodes in the graph. EGR is a robustness measure which picks up the nodes whose removal maximizes the decrease in graph robustness. Therefore, the use of these graph theoretic metrics as AL heuristics leads to an improvement in classification performance by upto 10% as compared to the baseline.

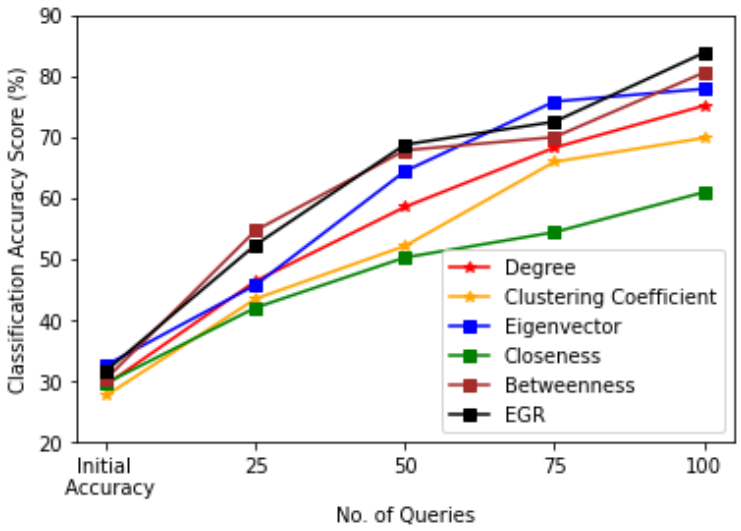


Figure 4.2: Plots of classification accuracy scores vs. no. of queries for Cora dataset.

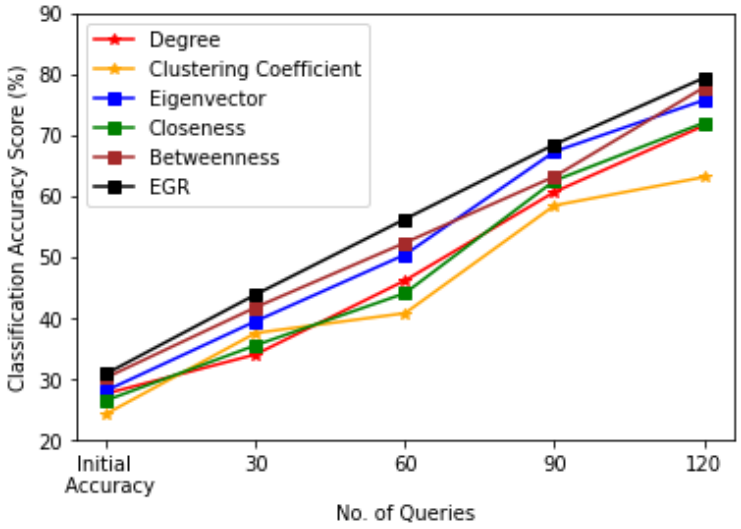


Figure 4.3: Plots of classification accuracy scores vs. no. of queries for CiteSeer dataset.

4.5 Summary

This chapter presents an AL framework to address node classification task in attributed graphs using a convex optimization approach. The proposed method integrates both topological information as well as node attributes within the learning scheme. The decision models are trained and updated using GraphSAGE, an inductive node embedding approach that learns both the topological structure and distribution of features for a node in its local neighborhood. Graph theoretic measures are used as AL heuristics to select the most informative nodes in the graph for annotation. It is observed that betweenness centrality and EGR consistently outperform the baseline and improve the classification performance by upto 10%. The next chapter focuses on representation of AL in the form of a collaborative decision environment of AI engines and human experts. The impacts of behavioral biases (associated with human experts) on the performance of AL strategies shall be studied.

Chapter 5

Impacts of behavioral biases on Active Learning

There is a plethora of literature published on handling practical AL challenges, like cold-start problem, oracle uncertainty, variable labelling costs and performance evaluation in the absence of ground truth. However, the collaboration of human and AI engine in a decision environment is neither straightforward nor well understood. There are anomalies and biases associated with both human and AI components of the decision environment. Algorithmic biases (like, selection bias, sampling bias, correlation fallacy, etc.) arises due to inability of algorithms to appropriately adjust to differences in data from different population subgroups [5]. On the other hand, behavioral biases (like, overconfidence, cognitive bias, hot hand fallacy, regret aversion bias, etc.) creep in due to uncertainties associated with human decisions [6].

This chapter demonstrates the simulation of different behavioral biases in an AL context. AL is represented as a collaborative decision environment consisting of AI engines (ML-based models) and human experts (Oracle). The user inputs are designed to be taken in two steps: agreement or disagreement with the AI model, followed by class labels based on human judgement (in case of disagreement). The behavioral biases are simulated by providing pre-engineered human decisions during the input steps. All the bias models are validated by

performing experiments on a real-world pancreatic cancer dataset. Further, the impacts of all the simulated biases on the performance of AL strategies are examined by comparing classification accuracy against an ideal case, which does not subsume any type of behavioral bias. It is observed that the accuracy score of the decision model is reduced by at least 20% (in cases of hot-hand fallacy and representative bias) to around 85% (in case of gambler's fallacy). Such a collaborative decision framework, with the flexibility to study multiple behavioral biases, has not been proposed in the literature and is a novel contribution of this work.

5.1 Related Work

Human experts are crucial components of AI-enabled services in Cyber-Physical-Human Systems (CPHS). They form a collaborative decision environment with the support of AI-based computational models. This is pertinent in a wide variety of domains, including fault diagnosis, predictive maintenance, optimal control, process and manufacturing industry operations and medical diagnosis. Given the compelling role of humans in such decision environments, it is an important research challenge to model, predict and use the limits of human behavior (e.g., behavioral bias and cognitive fatigue) in CPHS design [87]. Modeling human behavior in a decision environment is not straightforward. Humans use cognitive mechanisms and decision heuristics to process information and make decisions under uncertainty [88, 89].

Behavioral biases have been studied in numerous fields, including investment and finance [90], radiology [91], medical diagnosis [92], and human-in-the-loop systems [93]. The existence of behavioral biases for investment decisions is studied, supported by evidence from the Indian stock market [94]. Protte et al. [6] have presented the impacts of overconfidence bias and hot-hand fallacy with the help of an experimental framework involving surveillance drone piloting. Cognitive bias and carelessness are parameterized, and their impact on users' reliability is evaluated for personal context recognition [95]. Furthermore, several recent studies have proposed methodologies to address algorithmic biases using effective sampling approaches [96–99] and adversarial learning [100–102]. Among all these studies

presented in the literature, a generic framework for modelling and prediction of behavioral biases in the context of a collaborative decision environment has not been proposed so far. An interactive framework with the flexibility to simulate and predict different behavioral biases would be highly beneficial to study, analyze and use the limits of human decision under uncertainty in human-AI collaborative decision-making tasks.

5.2 Behavioral Bias Models

The irrational behaviors of humans which abstractly hinder the logical decision process are known as behavioral biases. The human decision or judgement methodically deviates from rationale, under the influence of these biases. This can lead to serious consequences, especially in domains like human health and medicine, where the stakes associated with decision-making are high. In this work, the following behavioral biases are considered: herding bias, cognitive bias, hot-hand fallacy, representative bias, anchoring bias, gambler's fallacy and regret-aversion bias.

In order to simulate the behavioral biases, the AL framework has been designed to query human experts in two steps:

(I) Firstly, the instance selected by the Active Learner is labelled as per the AI model trained at the current step. This instance is then presented to the human expert along with the predicted label, who is asked to specify whether he/she agrees or disagrees with the decision of the AI model.

(II) If the human expert agrees with the decision of AI model, the predicted label is considered to update the model, otherwise the human expert is prompted to provide a label as per his/her judgement.

In this work, each of the behavioral biases is simulated by supplying pre-engineered human decisions during both the input steps, based on the foundational understanding of respective biases. On the other hand, the human inputs corresponding to "Ideal Case" are formulated

based on the ground-truth labels in the dataset, which justifies the absence of behavioral biases.

Herding bias is the tendency of humans to take a specific decision just because it is being supported by many other people, rather than relying on their own judgement. This is simulated in our AL environment by making the human expert to indiscriminately agree with the AI model during step (I) of each query. Cognitive bias arises from the generation of a strong, falsified preconceived notion in human minds. Henceforth, there is a tendency to form mental shortcuts to process the information quickly, rather than making rational decisions. This is simulated by making the human expert to blindly disagree with the AI model during step (I) of the input process. Further, their decisions are simulated by supplying uniformly distributed random numbers as shown in (5.1) during step (II) of each query. Here, C is the number of classes and d_j is decision of the human expert at step (II) for jth query.

$$d_j \sim U(1, C) \tag{5.1}$$

Hot-hand fallacy causes humans to overconfidently believe that their decision will be correct based on sequences of immediate correct decisions in the past. This is a "fallacy" because a future outcome is independent of the past performance. This is simulated by considering ground-truth labels during an initial set of queries, similar to that in Ideal Case. After an initial set of queries, the inputs are formulated so as to make the human expert to always disagree with AI model in step (I) and generating uniformly distributed random numbers in step (II) to mimic the overconfidence effect in hot-hand fallacy. Representative bias leads to decisions being taken based on an erroneous prototype already existing in the human minds. This "prototype" is typically the most relevant example of a particular object or event. It results in overestimation of similarity between two things that are being compared by the humans. In the AL environment, ground truth labels are considered during initial fraction of queries. Once the representative bias sets in, the inputs in step (I) are designed to have the human expert randomly agree/disagree with the AI model, followed by uniformly distributed random numbers in step (II), as indicated in (5.1).

Anchoring bias induces the human decisions to over-rely on first piece of information about a particular event or object. This skews the human judgement and prevent them from making rational decisions. This is simulated in our AL environment by having inputs so as to make the human expert to always agree with the AI model after an initial set of queries. This emulates the decision of human experts to be anchored based on the information learn during initial queries. Gambler's fallacy causes humans to erroneously predict the probability of a random event based on the outcomes corresponding to sequences of immediate events in the past. Although the human expert would have made a series of incorrect decisions, he/she would still go ahead for another wrong decision overconfidently, in the hope of making a correct one. This is simulated by having the human experts to forcibly make wrong decisions, i.e., shuffling the ground-truth class labels for a fraction of instances in the query set.

Regret-aversion bias occurs when human experts make decisions, so as to avoid regretting alternate decisions in the future. Under the influence of this bias, the expert prefers to select the option that would carry the least regret, even if it is not the most appropriate choice. This is simulated in our AL environment by modifying the ground-truth labels to replace them with the ones corresponding to a pessimistic choice (for example, replacing the label corresponding to lower grade of a disease with the one corresponding to higher grade of the same) for a fraction of instances in the set of queries.

5.3 Experimental setup

In this chapter, simulation of all the behavioral biases discussed in Section 5.2 is demonstrated in an AL context, on a pancreatic cancer dataset adapted from [103]. It comprises of data from 159 participants, classified into 4 classes (healthy, pancreatitis, localized and metastatic) on the basis of an enzymatic signature consisting of arginase, matrix metalloproteinase-1, -3, and - 9, cathepsin-B and -E, urokinase plasminogen activator, and neutrophil elastase [103]. 10% of the total instances in the dataset are selected randomly to create an initial labeled dataset, which is used to train an initial ML-based classification model. Further, 50% of the total instances are used for querying iteratively (one query per

iteration), and the classification model is updated after each query step. kNN is chosen as the base classification method because it is versatile, simple and easy to implement and a non-parametric classification algorithm. Moreover, it does not make any inherent assumptions about the distribution of input data. US query strategy is implemented in Python 3.8 to select instances for annotation by the human experts. US selects instances for querying from the pool of unlabeled samples which minimizes the classifier uncertainty, as described mathematically in (5.2). Here, $\hat{y}$ is the predicted label for the instance x under the model θ. In the US query strategy, $\hat{y}$ is the prediction with the highest posterior probability under the model θ (indicated in eq. (5.3)) and x^* is the instance chosen for annotation by the human expert.

$$x^* = \arg\min_x P_\theta(\hat{y}|x) = \arg\max_x 1 - P_\theta(\hat{y}|x) \tag{5.2}$$

$$\hat{y} = \arg\max_y P_\theta(y|x) \tag{5.3}$$

5.4 Results

For the sake of convenience, the behavioral biases discussed in Section 5.2 are categorized into 4 categories: Representative bias and Anchoring bias are classified as *Chasing Trends*; Hot-hand and Gambler's fallacies are *Overconfidence* biases; Herding bias and Cognitive bias fall under *Limited Attention Span*; and *Regret-aversion* bias is treated as a separate category. In order to study the impacts of all these behavioral biases in AL setting, the performance (i.e., classification accuracy score) of the model is recorded after each query step for all the cases. The plots of accuracy scores for all categories of biases are presented in Figures 5.1, 5.2, 5.3 and 5.4.

It can be seen that the accuracy score increases with increase in no. of queries for the Ideal Case. The model trained with initial labelled dataset classifies around 65% of the instances correctly. This score gradually increases to around 96% after 80 queries are made

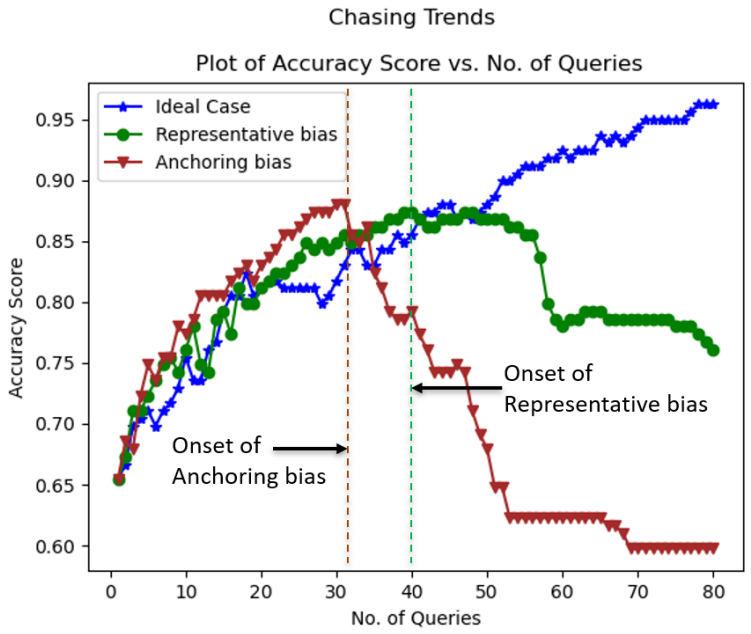


Figure 5.1: Plot of Accuracy Score vs. No. of Queries: Chasing Trends.

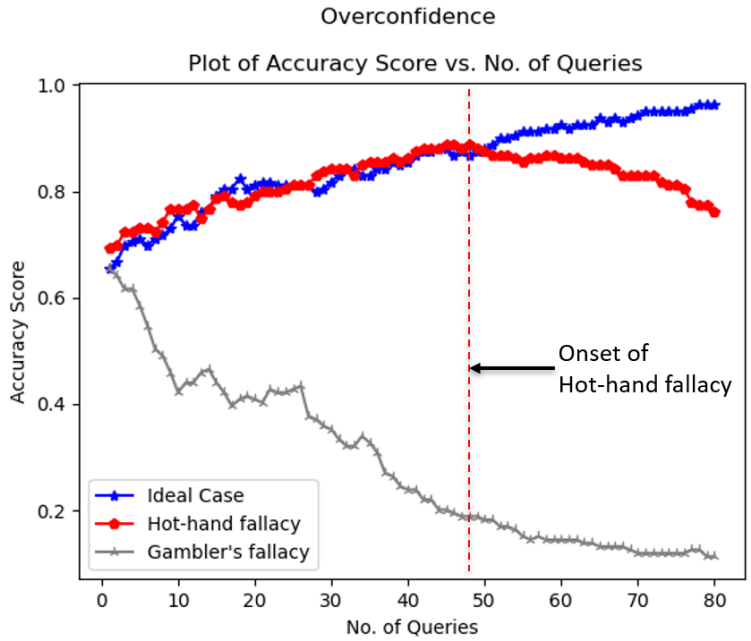


Figure 5.2: Plot of Accuracy Score vs. No. of Queries: Overconfidence.

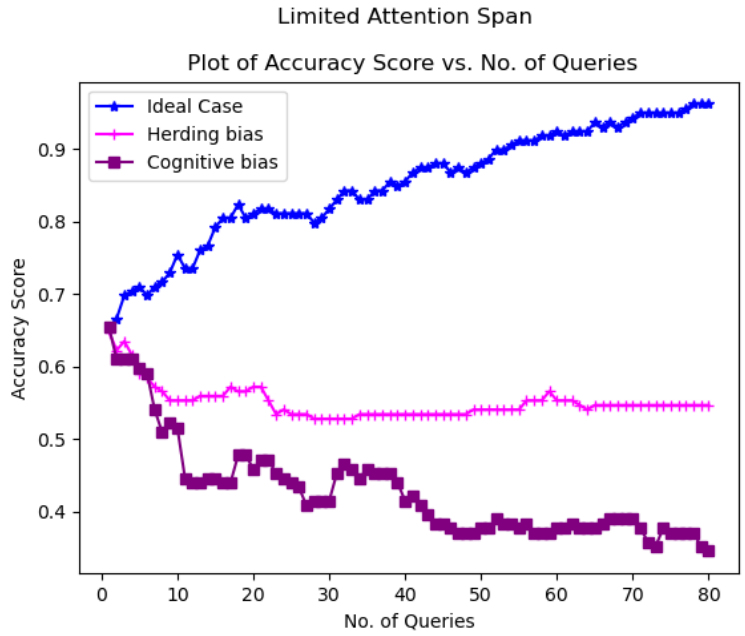


Figure 5.3: Plot of Accuracy Score vs. No. of Queries: Limited Attention Span.

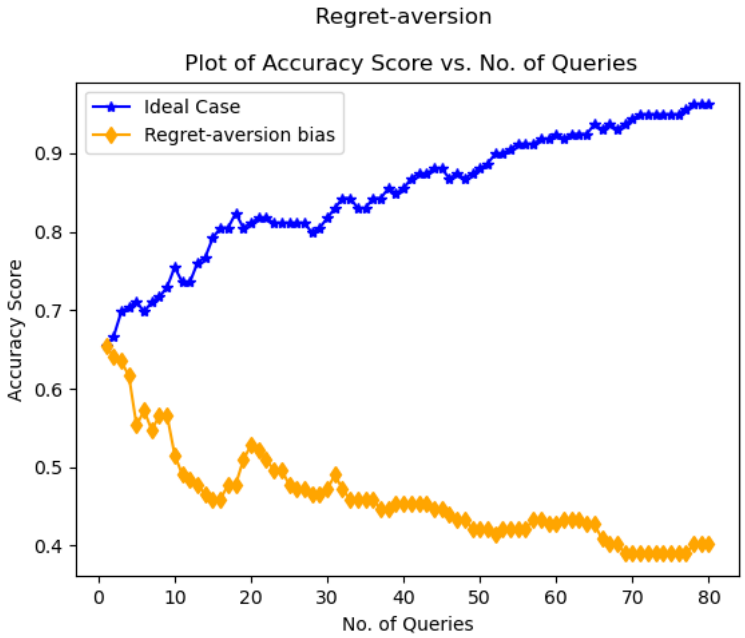


Figure 5.4: Plot of Accuracy Score vs. No. of Queries: Regret-aversion.

Table 5.1: Confusion matrix for Ideal Case after 50% queries

		Predicted Class			
		Healthy	**Pancreatitis**	**Localized**	**Metastatic**
True Class	**Healthy**	50	0	0	0
	Pancreatitis	2	23	0	1
	Localized	0	0	32	1
	Metastatic	0	2	0	48

to the human annotator and model being updated after each query step. The corresponding confusion matrix is shown in Table 5.1. However, this trend is not observed in case of any of the behavioral biases. For Representative bias (Figure 5.1), the accuracy score increases upto 40% of the queries. The inputs are provided so as to set in the Representative bias at this point. Once its sets in, the accuracy score reduces with increasing no. of queries. This is because the human decisions are biased due to an erroneous prototype already existing in their minds. Similarly, for the cases of Anchoring bias (Figure 5.1) and Hot-hand fallacy (Figure 5.2), the accuracy score start decreasing after the corresponding biases set in at 50% and 60% query steps respectively. Further, it can be seen that in the case of Cognitive bias (Figure 5.3), the accuracy score consistently reduces with increase in no. of queries. This is because the human experts make biased decisions due to a strong, falsified preconceived notions. They tend to form mental shortcuts for quick information processing, rather than making rational decisions. Similar trends can be observed for Gambler's fallacy (Figure 5.2), Herding bias (Figure 5.3) and Regret-aversion bias (Figure 5.4). In each of these cases, the human experts make decisions biased on several factors as discussed in Section 5.2, rather than relying on their own logical judgement.

5.5 Summary

In this chapter, the impact of seven behavioral biases, namely, herding bias, cognitive bias, hot-hand fallacy, representative bias, anchoring bias, gambler's fallacy and regret-aversion bias is illustrated using experiments conducted on a real-world pancreatic cancer dataset. Firstly, AL is represented in the form of a collaborative decision environment of AI engines and human experts, and the annotation by human experts is formulated as a two-step process.

Secondly, the behavioral biases are simulated by dispensing pre-manipulated user inputs based on the foundational understanding of respective biases during the iterative query steps. Finally, the impacts of these biases on the performance of AL strategies are assessed by comparing classification accuracy score of the decision model against a reference case, which does not assimilate any sort of behavioral bias. It is observed that the performance deteriorates significantly when the human decisions are influenced by each of the behavioral biases. The next chapter focuses on tracking and handling these behavioral biases in AL frameworks.

Chapter 6

Handling behavioral biases in Active Learning

This chapter presents a systematic framework for modeling, tracking and adaptation of behavioral biases in human-AI interactive decision environments within the context of AL. The specific contributions are as follows:

- The human-AI interactive decision environment is modeled in the form of a coupled dynamical system consisting of human experts and AI engines using communication graphs. This novel modeling approach helps characterize the dynamics of error variables that arise when the communication graph changes over iterative queries within the AL environment.

- For the first time, a novel Exploitation-Scrutinization-Investigation-Scrutinization (ESIS) strategy is proposed for tracking of behavioral biases within AL settings. Specifically, the querying process is strategically designed and split into four iterative stages to track the extent of behavioral biases and subsequently infer the grade of human expert.

- Once the behavioral biases are tracked, a unique model adaptation framework is introduced to handle these biases by assigning appropriate weights to samples during the iterative training procedure in AL. These weights are evaluated based on the values

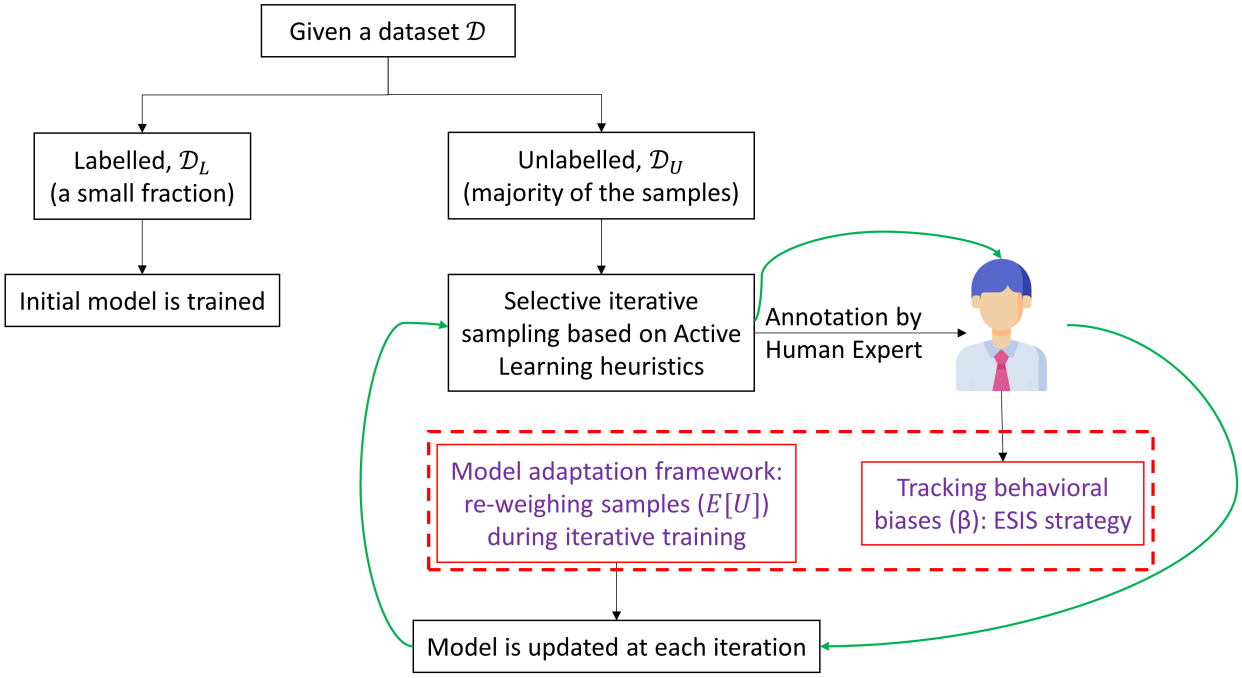


Figure 6.1: Proposed framework for tracking and handling behavioral biases in AL frameworks.

of expected collaborative utility function. The samples with higher value of expected collaborative utility are allotted higher weights during training and vice-versa.

Thus, the novelty of this work lies in adding two new components (one for tracking the behavioral biases and the other for subsequent model adaptation) in the traditional AL cycle. This is illustrated in Figure 6.1. The components in black outline represent the existing traditional AL frameworks. The components in red outline depict the novel contributions of this work. A scalar parameter β is introduced to track behavioral biases using a novel ESIS strategy, and these are consequently handled by attributing appropriate weights to the training samples based on expected collaborative utility function ($E[U]$). The green arrows indicate the iterative training procedure of the HITL decision system.

An approach for tracking of representative bias and subsequent model adaptation in this work. Behavioral biases are imitated by providing pre-engineered human decisions for both input AL steps (illustrated in Section 5.2), based on a fundamental understanding of respective biases. Representative bias causes humans to make decisions based on an

erroneous prototype that already exists in their minds [104]. This "prototype" is often the most representative instance of a specific item or occurrence. It causes humans to overestimate the similarity between two objects they are comparing. Ground truth labels are considered during the initial fraction of queries in the AL context. As discussed in Section 5.2, the inputs in step (I) are meant to make the human expert randomly agree/disagree with the AI model, followed by uniformly distributed random numbers in step (II). Although the demonstration of tracking and model adaptation strategies is limited to representative bias in this work, the proposed framework is generic and can be applied across any type of behavioral biases.

6.1 Related Work

As we look ahead, human experts may be a part of a collaborative decision environment with AI/ML-based computational models. Such synergistic frameworks are applicable across many critical domains like medical diagnosis, optimal control, predictive maintenance, fault monitoring and safety-critical procedures in manufacturing industries. Thus, modeling, predicting, and using the limits of human behavior in HITL systems is a prominent research challenge [87]. Behavioral biases associated with human behavior in decision making and algorithmic biases associated with supporting AI engines are key challenges in HITL systems.

Recently, numerous studies have proposed several methodologies to address algorithmic biases. These are handled primarily using two major techniques: effective sampling approaches [96–99] and adversarial learning [100–102]. The authors in [96] handle algorithmic biases by employing variational autoencoder to learn the latent structure within the data. These learned latent distributions are then used to re-weigh the training data points based on their importance. Iosifidis et al. [97] propose data augmentation techniques to deal with algorithmic biases at the input/data layer. Generative Adversarial Networks (GAN) are proposed to generate training data where the model fails [98]. This is coupled with population bias visualisation, that groups features by similarity and those instances are identified where the model fails to generalize. This procedure is continued iteratively till convergence.

A novel probabilistic formulation for data pre-processing is introduced to mitigate the algoritmic biases [99]. Alvi et al. [100] propose an algorithm to eliminate multiple sources of deviations from the feature representation of a neural network. This is then utilized to mitigate algorithmic biases from the feature representation while training networks on highly biased datasets and consequently improve the model performance. A domain-independent training technique based on adversarial learning is presented for mitigation of algorithmic biases [101]. In another study conducted by Zhang et al. [102], the algorithmic biases are handled by simultaneously learning a predictor and an adversary. The fundamental premise here is to maximize the predictor's ability to make predictions, while simultaneously minimizing the adversary's ability to model a protected feature/variable.

It is difficult to model human behavior in a decision-making context accurately. Humans analyze information and make decisions under ambiguity by employing cognitive mechanisms and decision heuristics [88, 89]. There exists a limited number of studies addressing behavioral biases. These biases have been investigated and analyzed for various applications like visual analytic systems [93], radiology [91], finance and investment [105, 106], as well as medical diagnosis [92]. The impacts of hot hand fallacy and overconfidence bias in the context of surveillance drone piloting have been studied with the help of an experimental setup by the authors in [6]. Giunchiglia et al. [95] have parameterized carelessness and cognitive bias, followed by the evaluation of users' reliability in reference with personal context recognition. A review of finance-related behavioral biases is provided and a framework for identifying behavioral biases in investment decisions has been proposed [105]. Another study focuses on the existence of behavioral biases for investment decisions [94], supported by the evidence from the Indian stock market. A fuzzy analytic hierarchical framework has been proposed to evaluate the effects of herding bias, loss aversion bias and overconfidence bias on decision-making of individual equity investors [106]. The influence of behavioral biases is studied in the context of entrepreneurial strategic decision-making [107]. However, all these works in the literature lack a generic framework for modelling, tracking and adaptation of behavioral biases. An interactive framework is proposed with the flexibility to simulate different behavioral biases in our work [10]. This work builds upon the interactive framework in [10] and

Figure 6.2: Communication graph between human expert and AI model in AL frameworks.

presents a novel strategy for tracking and subsequent model adaptation of behavioral biases. Additionally, these biases are modeled as a coupled dynamical system between human and AI agents via a communication graph in the context of HITL decision environments.

6.2 Modeling Behavioral Biases

The collaboration between humans and AI engines in a decision environment is not well studied in the literature. This is because the interaction between these agents is not straightforward and encompasses anomalies/biases associated with each of them. In this work, the human-AI collaborative decision environment is modeled as a coupled dynamical system consisting of human and AI components via a communication graph. This approach captures the dynamics of variables that arise when the communication graph changes over iterative queries within the AL environment. The interaction between human experts and AI agents in AL frameworks is represented in Fig. 6.2. The errors on human and AI ends are characterized by e_H and e_M respectively. e_H captures all the errors due to behavioral biases (like cognitive bias, overconfidence, regret aversion, hot hnd fallacy, etc.) that emerge as a result of uncertainties associated with human decisions [6]. On the other hand, e_M captures all the errors arising due to classification inaccuracies of the AI model. The human-AI and AI-human link weights are indicated by w_{HM} and w_{MH} respectively. w_{HM} is dictated by the grade of human expert, which can be quantified using numerous ways. It is inversely proportional to the expertise level of the human expert, i.e., $w_{HM} \propto 1/(\text{expertise level})$. In this work, three grades of human experts are considered: high, medium and low. Hence, one

possible representation of w_{HM} is given by

$$w_{HM} = \begin{cases} \frac{1}{3}, & \text{for high-grade expert } (\Lambda_1) \\ \frac{2}{3}, & \text{for medium-grade expert } (\Lambda_2) \\ 1, & \text{for low-grade expert } (\Lambda_3) \end{cases} \tag{6.1}$$

It should be noted that the definition of human experts grades illustrated in eq. (6.1) is just an example and there can be many other ways to outline the same. The proposed framework is generic and is applicable across all such divisions pertaining to the grades of human experts. w_{MH} is guided by a scrutinization score, which is a part of the ESIS bias tracking strategy described in Section 6.3.1. At any given query step in AL, scrutinization is performed by considering the performance of the decision model over a predefined window of the last few queries. That is,

$$w_{HM} = -\frac{v[q]}{n_s} \tag{6.2}$$

where, n_s represents the number of queries considered in the scrutinizing window; the scrutinization score at query step q (i.e., $v[q]$) is defined as

$$v[q] = \sum_{i=0}^{n_s-1} c[q-i] \tag{6.3}$$

$$c[a] = \begin{cases} +1, & \text{when } J[a] > J[a-1] \\ 0, & \text{when } J[a] = J[a-1] \\ -1, & \text{when } J[a] < J[a-1] \end{cases} \tag{6.4}$$

$J[a]$ denotes the value of performance measure (like, classification accuracy) of the decision model at query step a. $v[q]$ is increased by 1 when performance improves, decreased by 1 when performance degrades and remains unchanged when performance remains constant over each of the iterative queries in AL environment. Again, this definition of scrutinization

score is just an illustration and can vary based on the structure of the decision environment and query settings. The proposed framework is capable to incorporate any mathematical formulation of the scrutinization score.

The primary goal of analyzing this information flow between the human and AI agents is to understand the dynamics of variables e_H and e_M, as the communication graph changes over iterative queries in AL environment. Therefore, the values of these variables need to be updated at each query step. In order to incorporate the dynamics of both human as well as AI agents within the framework, the values of e_H and e_M at subsequent query steps $(q+1)$ should individually account for both e_H and e_M at the previous query step q. This is mathematically described as

$$e_H[q+1] = f_{\underline{\theta_1}}(e_H[q], e_M[q]) \tag{6.5}$$

$$e_M[q+1] = g_{\underline{\theta_2}}(e_H[q], e_M[q]) \tag{6.6}$$

Here, $\underline{\theta_1}$ and $\underline{\theta_2}$ are the predefined or learnable parameters. For the sake of notational simplicity, $\underline{\theta_1}$ and $\underline{\theta_2}$ are not included during the follow-on discussion in this work. Thus, eqns. (6.5) and (6.6) can be rewritten as

$$e_H[q+1] = f(e_H[q], e_M[q]) \tag{6.7}$$

$$e_M[q+1] = g(e_H[q], e_M[q]) \tag{6.8}$$

Definition 2. *In a human-AI collaborative decision environment represented as a communication graph between human and AI agents, the errors corresponding to both the components (indicated by e_H and e_M respectively) are updated over iterative AL queries as described in eqns. (6.7) and (6.8), with $f(\cdot)$ and $g(\cdot)$ satisfying the following properties:*

(i) ***Non-negative:*** $f(x, y) \geq 0$; $g(x, y) \geq 0$

(ii) ***Commutative:*** $f(x,y) = f(y,x)$ *and* $g(x,y) = g(y,x)$

(iii) ***Bounds:*** $\min(x,y) \leq f(x,y), g(x,y) \leq \max(x,y)$

In a classification problem, the errors e_H and e_M are dictated by the number of misclassifications in the model predictions, which is always a non-negative quantity. Assuming the error to be absolute difference of predicted and actual output quantities for any regression problem, the property *(i)* should always hold true. Every iteration in AL frameworks involve contributions from both the components, i.e., human expert as well as AI agent. The HITL settings might involve contributions from individual components in any order, i.e., input from humans followed by deliberation by an AI agent [108], [109], or contemplation by the AI agent followed by opinions of human experts [10], [110]. It is to be noted that that the "order" over here refers to the sequence of inputs obtained from human expert and AI agent within a single AL iteration and does not incorporate any time-related aspects of the AL framework. The proposed framework is capable of incorporating both these scenarios and the evolution of errors e_H and e_M over different queries should be independent of the sequence of these operations within the human-AI collaboration, thereby holding property *(ii)*. It is well established that the collaborative efforts of human and AI agents leads to an enhanced performance, thereby reducing the errors associated with individual components [111], [112]. Moreover, AL frameworks lead to improvement in model performance (reduction in errors) as more queries are made during the iterative querying setup [10]. This justifies the maximum bound of $f(\cdot)$ and $g(\cdot)$ functions in property *(iii)*. However, the overall model performance in AL frameworks is limited either by annotation budget in terms of labeling cost and time [113], or due to behavioral and cognitive factors associated with human experts [10]. This accounts for the minimum bound of $f(\cdot)$ and $g(\cdot)$ functions in property *(iii)*.

In this work, the values of e_H and e_M are updated by taking the weighted averages (weighted by corresponding human-AI or AI-human link weights in the communication graph). This can be represented as

$$e_H[q+1] = \frac{e_H[q] + w_{MH} e_M[q]}{1 + w_{MH}} \tag{6.9}$$

$$e_M[q+1] = \frac{e_M[q] + w_{HM} e_H[q]}{1 + w_{HM}} \tag{6.10}$$

For this specific case, $\underline{\theta_1}$ and $\underline{\theta_2}$ in eqns. (6.5) and (6.6), respectively, can be represented as: $\underline{\theta_1} = w_{MH}$ and $\underline{\theta_2} = w_{HM}$. It is trivial to show that the weighted average fucntion satisfies properties *(i)-(iii)* illustrated in Definition 2. Eqns. (6.9) and (6.10) reinforce the following intuitive arguments: (i) the high-grade experts have lower contributions towards increase in model inaccuracies, as compared to medium-grade and low-grade experts, (ii) a poor scrutinization score leads to an increase in behavioral biases, whereas a good scrutinization score reduces the behavioral biases. A reduction in behavioral biases signifies improvement in the grade of human experts.

The overall discrete-time coupled dynamical system comprising of human expert and AI engine corresponds to

$$e[q+1] = Te[q], \tag{6.11}$$

where,

$$e[q+1] = \begin{bmatrix} e_H[q+1] \\ e_M[q+1] \end{bmatrix}, \tag{6.12}$$

$$e[q] = \begin{bmatrix} e_H[q] \\ e_M[q] \end{bmatrix}, \tag{6.13}$$

and

$$T = \begin{bmatrix} \frac{1}{1+w_{MH}} & \frac{w_{MH}}{1+w_{MH}} \\ \frac{w_{HM}}{1+w_{HM}} & \frac{1}{1+w_{HM}} \end{bmatrix} \tag{6.14}$$

6.3 Tracking and Model Adaptation of Behavioral Biases

This work presents an integrated framework for tracking and incorporating behavioral biases within AL settings. Firstly, a novel ESIS strategy is proposed for tracking of behavioral biases. Once tracked, these biases are handled by assigning appropriate weights to the individual samples during iterative training process in AL. The training weights associated with each of the samples are evaluated based on the values of expected collaborative utility function. Each of these components are discussed in the following subsections.

6.3.1 Tracking of behavioral biases

A novel ESIS strategy (summarized in Figure 6.3) is proposed for tracking behavioral baises in AL frameworks. The querying process is strategically designed and split into four iterative stages (exploitation, scrutinization, investigation and scrutinization) to track the extent of behavioral biases and subsequently infer the grade of human expert. Typically, the decisions taken by higher grade of human experts are considered to be less biases than those of lower grade experts. This is because the higher grade experts shall have better domain knowledge and experience, which facilitates superior decision making abilities as compared to lower grade experts [114]. There are several ways to quantify the extent of behavioral biases across different grades of human experts. In this work, a scalar parameter β is introduced to track biases. In general, this parameter can be probabilistically modeled as: $\beta \sim \mathcal{F}(\Lambda)$, where $\mathcal{F}$ is a distribution with parameter Λ. The parameter Λ can be used to reflect the level of expertise of the human. For example, $\Lambda \in \Lambda_1$ for high-grade expert; $\Lambda \in \Lambda_2$ for medium-grade expert; and $\Lambda \in \Lambda_3$ for low-grade expert, and $\Lambda_1 \cap \Lambda_2 \cap \Lambda_3 = \phi$. As an example of such a model, a uniform distribution is considered for β with $\Lambda_1 = \left(0, \frac{1}{3}\right)$; $\Lambda_2 = \left(\frac{1}{3}, \frac{2}{3}\right)$ and $\Lambda_3 = \left(\frac{2}{3}, 1\right)$. That is,

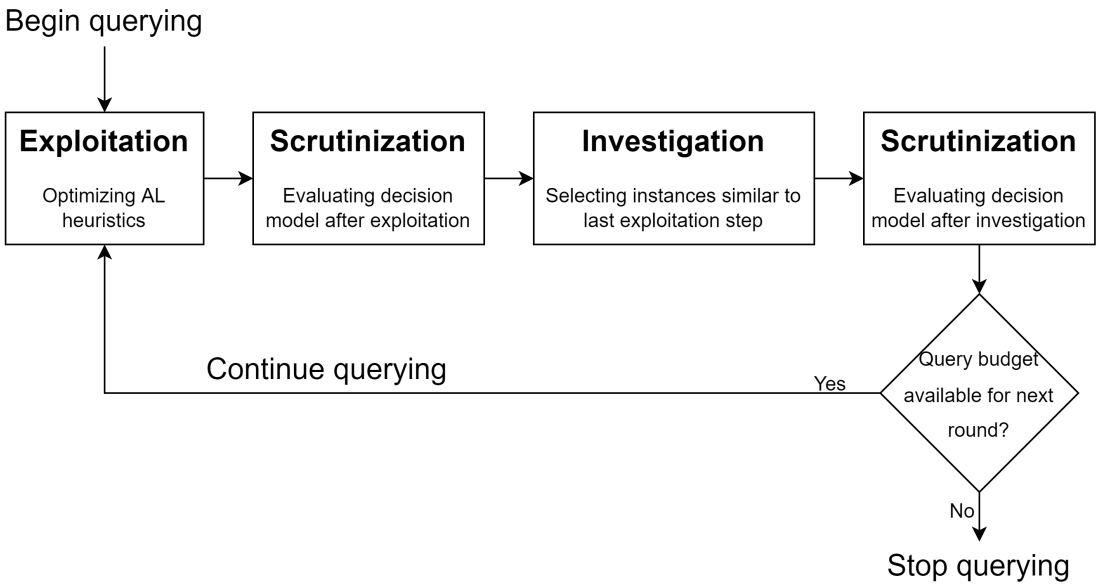


Figure 6.3: ESIS strategy for tracking behavioral biases.

$$\beta \sim \begin{cases} U\left(0, \frac{1}{3}\right), & \text{for high-grade expert} \\ U\left(\frac{1}{3}, \frac{2}{3}\right), & \text{for medium-grade expert} \\ U\left(\frac{2}{3}, 1\right), & \text{for low-grade expert} \end{cases} \tag{6.15}$$

where, $U(u_1, u_2)$ specifies uniform distribution in the range u_1 to u_2. While $\beta = 0$ denotes the case with no behavioral biases, $\beta = 1$ represents the decision heavily corrupted by behavioral biases. It is worthwhile to note that the specification of β in eq. (6.15) is just an example considered in this work. It can vary based on the experimental settings and design of the decision environment. The framework for modeling, tracking and adaptation of behavioral biases proposed in this work is generic and applicable across any specification of β, without being restricted to just the uniform distribution.

The querying process begins with exploitation steps for an initial set of queries (n_{se}). During these queries, the most informative instances are selected by optimizing AL heuristics (for example, uncertainty sampling, classification margin, or entropy [53, 81]), as in a typical AL framework. For instance, if the base query strategy is uncertainty sampling, the instances selected for annotation during the exploitation steps are given by:

$$x^* = \arg\min_x P_\theta(\hat{y}|x) \tag{6.16}$$

where,

$$\hat{y} = \arg\max_y P_\theta(y|x) \tag{6.17}$$

Here, $\hat{y}$ is the predicted label for instance x under the model θ, y is the true label and $P_\theta(\cdot)$ indicates the probability of a specific event under the model θ.

After n_{se} exploitation queries, scrutinization is performed to evaluate the change the performance of decision model over this window.

$$v[q] = \sum_{i=0}^{n_{se}-1} c[q-i] \tag{6.18}$$

This is followed by n_{si} investigation steps, where the instances similar to that selected during the last exploitation step are picked up for annotation by the human expert. This is indicated as follows:

$$x' = \arg\max_x T(x, x^*) \tag{6.19}$$

where, $T(x, x^*)$ can be any similarity measure like Cosine similarity, Euclidean distance, or Manhattan distance. The fundamental premise of having additional investigation queries is to ensure that the model performance obtained during exploitation is actually because of marginal learning and not at random by chance.

Finally, scrutinization is performed once again to capture the change the performance of decision model after the investigation steps.

$$v[q] = \sum_{i=0}^{n_{si}-1} c[q-i] \tag{6.20}$$

This ESIS querying cycle is repeated until the query budget is available and the decision model is updated at each query step. Furthermore, the value of β is updated at each scrutinization step based on the values of $v[q]$, as demonstrated in eq. (6.21).

Table 6.1: Utility Function

Meta-decision (m)	**Prediction**	
	Correct	**Incorrect**
Agreement (A)	$1-\lambda_A$	$-\lambda_A-\beta-\gamma$
Disagreement (D)	$1-\lambda_D$	$-\lambda_D-\beta-\gamma$

$$\beta_{q+1} = \begin{cases} \beta_q + \delta\beta, & \text{when } v[q] < 0 \text{ (degrading performance)} \\ \beta_q, & \text{when } v[q] = 0 \text{ (consistent performance)} \\ \beta_q - \delta\beta, & \text{when } v[q] > 0 \text{ (improving performance)} \end{cases} \tag{6.21}$$

The value of β at query $(q+1)$ (i.e., β_{q+1}) is obtained by: (1) increasing the value of β at previous query step (i.e., β_q) by a small amount $(\beta_q + \delta\beta)$ for degrading performance (indicated by $v[q] < 0$); (2) reducing β_q by a small amount $(\beta_q - \delta\beta)$ for improving performance (indicated by $v[q] > 0$); and (3) keeping β_q unchanged in case of consistent performance (indicated by $v[q] = 0$). The decreasing values of β denotes reduction in behavioral biases (leading to improvement in overall performance of the decision model) and vice-versa.

6.3.2 Model adaptation of behavioral biases

After tracking of behavioral biases using the proposed ESIS strategy, the effects of these biases are mitigated by assigning appropriate weights to the individual samples during iterative training process in AL. These weights are evaluated based on the values of expected collaborative utility function. Given the two-step input design of AL framework considered in this work, a utility function can be formulated as shown in Table 6.1. Here, λ_A and λ_D are costs incurred in additional analysis carried out by human experts, β and γ capture errors due to behavioral biases and model inaccuracies respectively.

The confidence of the model for a prediction is given by:

$$P(\hat{y} = y | m = A) = h(x)[\hat{y}] \tag{6.22}$$

Here, $h(x)[\hat{y}]$ represents the values of class probability corresponding to the predictions $\hat{y}$. The grade of the human experts (g) can be represented as:

$$P(\hat{y} = y|m = A) = g \tag{6.23}$$

The threshold confidence for agreement/disagreement of human expert with the recommendation of AI model (step (I) of the AL input design as described in Section 5.2) is formalised as Lemma 1.

Lemma 1. *The minimum confidence level for agreement of human expert with the recommendation of AI model is given by:*

$$s(\lambda_A, \lambda_D, \beta, \gamma, g) = g - \frac{(\lambda_D - \lambda_A)}{(1 + \beta + \gamma)} \tag{6.24}$$

where, λ_A and λ_D are costs incurred in additional analysis carried out by human experts; β and γ capture errors due to behavioral biases and model inaccuracies, respectively; and g represents the grade of the human expert.

Consequently, the policy of agreement is specified as:

$$P(m = A) = \begin{cases} 1, & \text{if } h(x)[\hat{y}] \geq s(\lambda_A, \lambda_D, \beta, \gamma, g) \\ 0, & \text{otherwise} \end{cases} \tag{6.25}$$

where, $\hat{y}$ is the predicted label for instance x.

Proof. The human expert agrees with the decision of AI model only if the expected utility for $(m = A)$ is greater than or equal to that for $(m = D)$.

$$E\left[U(m = A)\right] \geq E\left[U(m = D)\right], \tag{6.26}$$

where, $E[\cdot]$ represents the expected value.

$$E\left[U(m = A)\right] = h(x)[\hat{y}](1 - \lambda_A) - (1 - h(x)[\hat{y}])(\lambda_A + \beta + \gamma) \tag{6.27}$$

Simplifying (6.27),

$$E\left[U(m=A)\right] = h(x)[\hat{y}](1+\beta+\gamma) - \lambda_A - \beta - \gamma \tag{6.28}$$

$$E\left[U(m=D)\right] = g(1-\lambda_D) - (1-g)(\lambda_D + \beta + \gamma) \tag{6.29}$$

Further simplifying (6.29),

$$E\left[U(m=D)\right] = g(1+\beta+\gamma) - \lambda_D - \beta - \gamma \tag{6.30}$$

Substituting eqs. (6.28) and (6.30) in (6.26),

$$h(x)[\hat{y}] \geq g - \frac{(\lambda_D - \lambda_A)}{(1+\beta+\gamma)} \tag{6.31}$$

The minimum confidence level for agreement is given by:

$$s(\lambda_A, \lambda_D, \beta, \gamma, g) = g - \frac{(\lambda_D - \lambda_A)}{(1+\beta+\gamma)} \tag{6.32}$$

Therefore, from eqs. (6.31) and (6.32), the policy of agreement is specified as:

$$P(m=A) = \begin{cases} 1, & \text{if } h(x)[\hat{y}] \geq s(\lambda_A, \lambda_D, \beta, \gamma, g) \\ 0, & \text{otherwise} \end{cases} \tag{6.33}$$

□

The following result formalises the evaluation of expected collaborative utility of the coupled dynamical system consisting of human expert and AI agent.

Theorem 2. *The expected collaborative utility of the coupled dynamical system consisting of*

human expert and AI agent is given by:

$$E[U] = \begin{cases} h(x)[\hat{y}](1+\beta+\gamma) - \lambda_A - \beta - \gamma, & \textit{if } h(x)[\hat{y}] \geq s(\lambda_A, \lambda_D, \beta, \gamma, g) \\ g(1+\beta+\gamma) - \lambda_D - \beta - \gamma, & \textit{otherwise} \end{cases} \tag{6.34}$$

where, λ_A, λ_D, β, γ, $\hat{y}$, x, and g are as defined in Lemma 1.

Proof. The expected collaborative utility is calculated as:

$$\begin{aligned} E[U] =& P(m=A)[P(\hat{y}=y|m=A)(1-\lambda_A) + P(\hat{y}\neq y|m=A)(-\lambda_A-\beta-\gamma)] \\ &+ P(m=D)[P(\hat{y}=y|m=D)(1-\lambda_D) + P(\hat{y}\neq y|m=D)(-\lambda_D-\beta-\gamma)] \end{aligned} \tag{6.35}$$

Simplifying (6.35), we get

$$\begin{aligned} E[U] =& P(m=A)[h(x)[\hat{y}](1+\beta+\gamma) - \lambda_A - \beta - \gamma] \\ &+ (1-P(m=A))[h(x)[g(1+\beta+\gamma) - \lambda_D - \beta - \gamma] \end{aligned} \tag{6.36}$$

$$\begin{aligned} E[U] =& P(m=A)[(1+\beta+\gamma)[h(x)[\hat{y}] - g] + (\lambda_D - \lambda_A)] \\ &+ g(1+\beta+\gamma) - \lambda_D - \beta - \gamma \end{aligned} \tag{6.37}$$

Substituting the policy of agreement from Lemma 1,

$$E[U] = \begin{cases} h(x)[\hat{y}](1+\beta+\gamma) - \lambda_A - \beta - \gamma, & \text{if } h(x)[\hat{y}] \geq s(\lambda_A, \lambda_D, \beta, \gamma, g) \\ g(1+\beta+\gamma) - \lambda_D - \beta - \gamma, & \text{otherwise} \end{cases} \tag{6.38}$$

□

During the process of iterative querying and training in AL frameworks, the samples can be weighted based on the values of $E[U]$. The samples with higher $E[U]$ should be allotted higher weight during training and vice-versa. In this framework, the samples with

lower values of $E[U]$ might be either misclassified by the decision model or biased by human behaviour like overconfidence or hot-hand fallacy. These might also be outliers or possess heavy cost for human annotation.

The performance of decision model can then be computed as follows:

$$R = \frac{1}{N} \sum_{i=1}^{N} w_i \, \mathcal{L}(y_i, \hat{y}_i) \tag{6.39}$$

where, w_i are weights evaluated on the basis of values of $E[U]$, N is the total number of samples, and $\mathcal{L}(\cdot)$ is a loss function. $\mathcal{L}(\cdot)$ is a measure to determine the deviation between the current output of a decision model and the expected output. It defines how well the algorithm is able to model the given dataset.

6.4 Experimental Evaluation

In this work, the approach for tracking of behavioral biases and the subsequent model adaptation within AL settings is demonstrated using a pancreatic cancer dataset derived from [103]. The pancreatic cancer dataset captures protease/arginase activity measured using ultrasensitive nanobiosensors and consists of a set of eight biomarkers including arginase, matrix metalloproteinase-1, -3, and - 9, cathepsin-B and -E, urokinase plasminogen activator, and neutrophil elastase. The publicly available NCBI Gene Expression Omnibus (GEO) database was used to collect the identified biomarkers. Proteases that exhibited significantly different expression levels in two samples - a primary tumor sample and a sample of healthy tissue - had to both be from *Homo sapiens* and were therefore considered to be biomarkers. The gene expression analysis using data gathered from Gene Symbol, Entrez Gene ID, NCBI GEO and Unigene ID was employed to compute the features corresponding to a panel of seven proteases and arginase for each sample in the dataset. Human serum samples were obtained from the Biospecimen Repository Facility in the Cancer Center of the University of Kansas Medical Center. Protease/arginase activity was measured each for identified biomarker in all these samples after 60 minutes incubation. The detection of pancreatic

cancer is modeled as a multi-class classification problem as in [115]. The dataset consists of samples classified into 4 categories: "Healthy", "Pancreatitis", "Localized" pancreatic cancer and "Metastatic" pancreatic cancer. "Healthy" denotes the data corresponding to healthy volunteers. "Pancreatitis" indicates the ones with inflammation in pancreas. While "Localized" pancreatic cancer samples indicate the absence of any signs that the disease has spread outside the pancreas, "Metastatic" pancreatic cancer implies that the cancer has spread to other parts of the body too.

10% of the total instances in the dataset are used to create an initial labeled dataset. This is then used to train an initial ML-based model for AL framework to classify the health status of patients. Once an initial model is trained, the querying process is initiated. Fifty percent of the total instances are used for querying the instances iteratively (one query per iteration), and the classification model is updated after each query step. kNN is used as the base classification method because it is simple, easy to implement and versatile. Additionally, it does not entail assumptions pertaining to the distribution of input data. Uncertainty Sampling (US) query strategy is employed to select instances for annotation during the iterative querying process in AL environment. It is worthwhile to note that the use of kNN is just an illustration and the proposed framework is applicable across any base classification method. In this strategy, the instances which minimize the classifier uncertainty are selected for annotation during each query step, as indicated in eq. (6.16).

Five exploitation queries (n_{se}) and five investigation queries (n_{si}) are considered iteratively, as part of the ESIS strategy for tracking of behavioral biases. Further, the value of β is modified by adding (in case of improving performance during scrutinization step) or subtracting (in case of degrading performance during scrutinization step) $\delta\beta$ from the value of β at the previous query step. In this work, $\delta\beta$ is picked to be a very small value, i.e., 0.001. The following five different learning scenarios are simulated: (i) learning for low-grade expert; (ii) transition from low-grade to medium-grade expert; (iii) learning for medium-grade expert; (iv) transition from medium-grade to high-grade expert; and (v) learning for high-grade expert. The values of classification accuracy scores as well β over different queries are recorded and analyzed for all the learning scenarios. The variations in classification accuracy

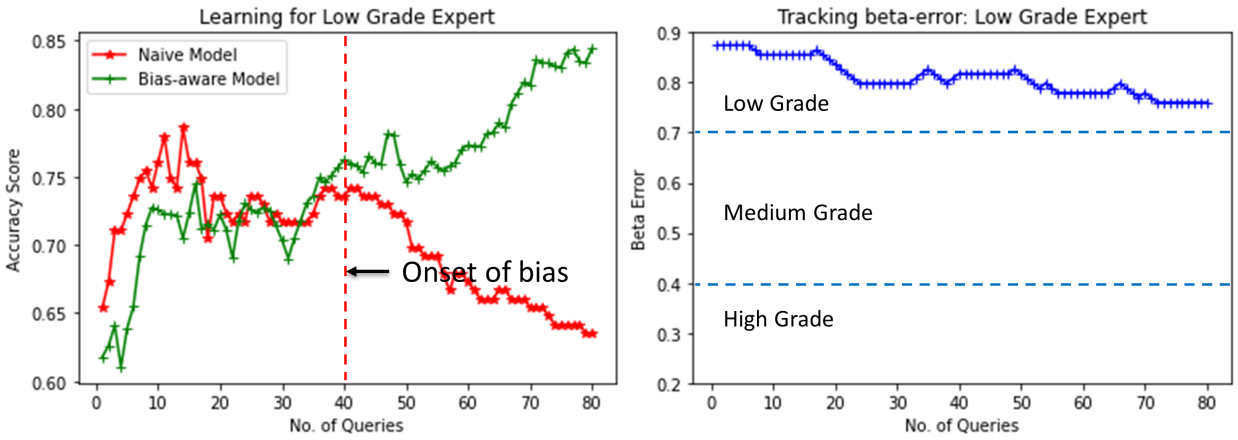


Figure 6.4: Plots for changes in classification accuracy and β: Scenario 1.

scores are compared in three cases: (i) decision models without the use of AL methods; (ii) naive decision models; and (iii) bias-aware decision models. When AL methods are not used, training instances are chosen randomly from the dataset, rather than strategic sampling by optimizing AL heuristics (as in eqns. (17) and (18)). Naive models employ AL methods, but do not incorporate knowledge of any existing behavioral biases during the learning process. On the other hand, bias-aware models adapt based on the behavioral biases, as discussed in Section 6.3.2. All the learning scenarios are discussed as follows.

- **Scenario 1: Learning for low-grade expert**

  In this scenario, the low-grade expert learns during the iterative querying process and the overall classification accuracy increases as more queries are made. The plots for changes in classification accuracy and β over different queries are presented in Figure 6.4. The decrease in values of β also reinforces the argument that the learning process leads to improvement in the performance of decision model. However, the learning in this scenario is not that significant to improve the overall grade of the human expert.

- **Scenario 2: Transition from low-grade to medium-grade expert**

  As opposed to Scenario 1, the learning in this scenario is prominent enough to upgrade the human expert from low-grade to medium-grade. The plots for changes in classification accuracy and β over different queries are presented in Figure 6.5. It can be seen

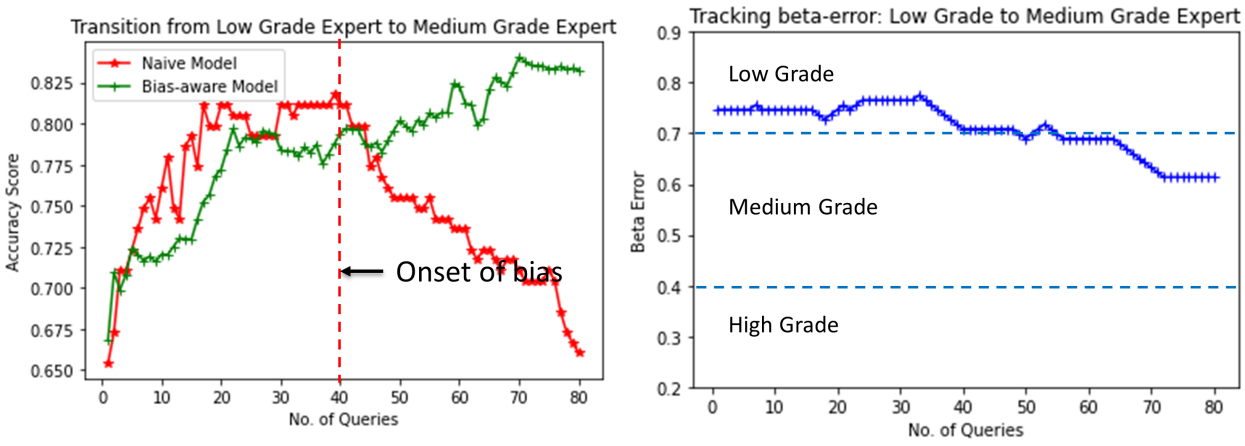


Figure 6.5: Plots for changes in classification accuracy and β: Scenario 2.

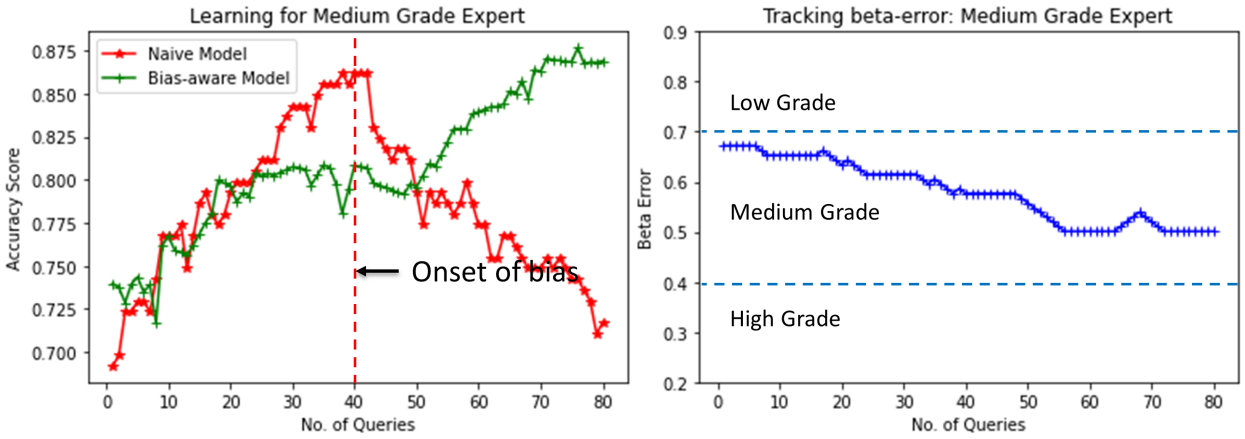


Figure 6.6: Plots for changes in classification accuracy and β: Scenario 3.

that the value of β crosses the threshold of 0.7, as discussed in eq. (6.15).

- **Scenario 3: Learning for medium-grade expert**

  This learning scenario is similar to Scenario 1 with low-grade expert replaced by the medium-grade expert. Figure 6.6 shows the plots for values of classification accuracy and β over different queries. It can be observed that the values of classification accuracy increases and that of β decreases as more queries are made to the human expert. Although the learning process improves the quality of decision model, it is not substantial enough to shift the human expert to next higher grade.

- **Scenario 4: Transition from medium-grade to high-grade expert**

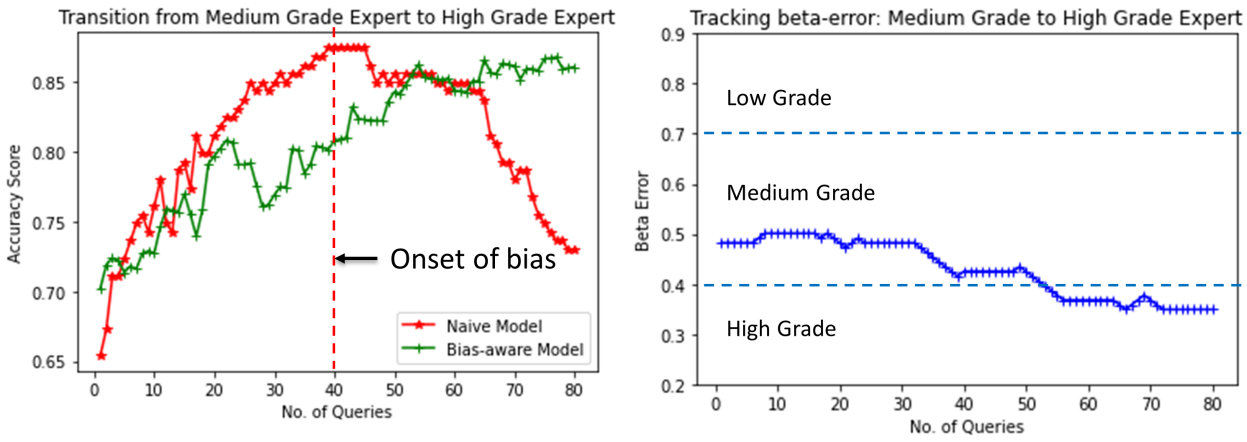


Figure 6.7: Plots for changes in classification accuracy and β: Scenario 4.

In this scenario, the learning process causes the human expert to upscale from medium-grade to high-grade. The changes in classification accuracy and β values are illustrated in plots provided in Figure 6.7. The classification accuracy is observed to improve with increasing number of queries. The value of β reduces and crosses the threshold of 0.4 (refer eq. (6.15)) as more instances are queried from the unlabeled pool of data.

- **Scenario 5: Learning for high-grade expert**

  This scenario illustrates learning for a high-grade expert during the iterative querying process in AL. Although the human expert involved in this scenario belongs to high-grade category, the learning process still helps them to further advance their decision-making capabilities. This is shown by decreasing values of β in Figure 6.8, as more queries are made. The increase in values of classification accuracy also indicate the improvement in performance of decision model as the learning progresses.

The performance comparison of the proposed bias-aware decision models with baselines (i.e., naive decision models) as well as decision models without the use of AL methods is summarized in Table 6.2. Firstly, it can be clearly established that the use of AL methods exhibit a significant advantage in performance of decision models, as compared to the case when AL methods are not employed. This is because AL allows strategic selection of training instances by optimizing AL heuristics, rather than random sampling. Further, it can be

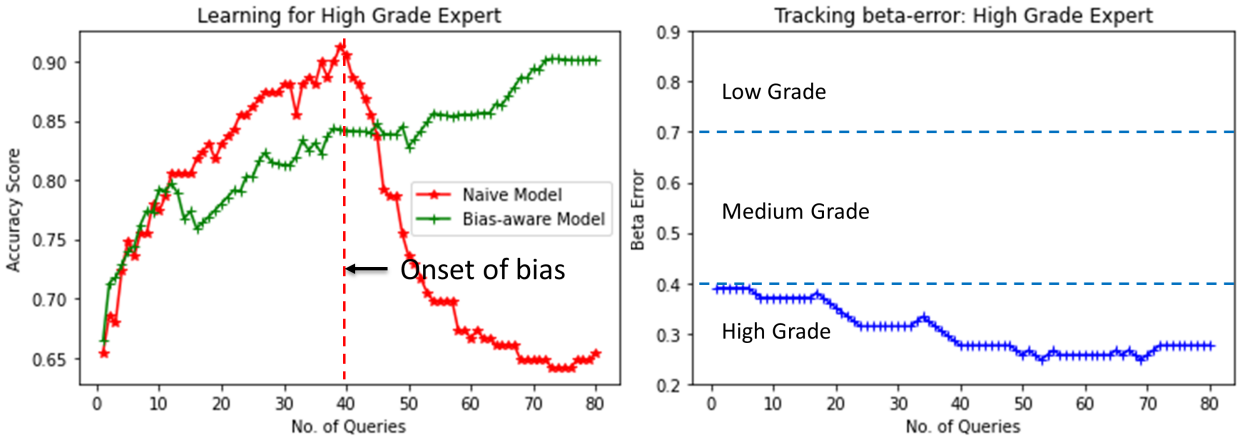


Figure 6.8: Plots for changes in classification accuracy and β: Scenario 5.

Table 6.2: Performance comparison of proposed bias-aware decision model with naive decision model (baseline) and without using AL methods.

Scenario	**Description**	**Classification Accuracy of AL models after 80 queries**		**Decision Models without AL**
		Naive Decision Model (Baseline)	**Bias-aware Model (Ours)**	
1	Learning for low-grade expert	64.83%	84.81%	61.66%
2	Transition from low-grade to medium-grade expert	65.38%	84.93%	62.41%
3	Learning for medium-grade expert	72.46%	86.62%	60.19%
4	Transition from medium-grade to high-grade expert	73.17%	87.03%	64.56%
5	Learning for high-grade expert	66.72%	90.15%	65.95%

observed that the classification accuracy of naive decision models start deteriorating once the effect of behavioral biases creep in. On the contrary, the classification accuracy of bias-aware decision models do not show significant decline. During the iterative querying process, the bias-aware decision models are consistently updated to accommodate the effects of behavioral biases that are tracked using proposed ESIS strategy. Despite the presence of behavioral biases, the bias-aware models exhibit an improving trend of performance as more queries are made within the learning framework.

All the learning scenarios discussed in this work portray a systematic progression of human expert from low-grade to medium-grade. While the overall performance of decision model is characterized by the classification accuracy score, β indicates the expertise level of the human expert at any given point of time. In all the learning scenarios, the bias-aware decision models are observed to deliver better performance as compared to naive decision models, at the end of querying procedure in the AL framework. It is observed that bias-aware models improve the final classification accuracy of naive decision models (at the last query step in AL) by upto 16% (in Scenario 5). This is because the bias-aware decision models continuously adapt based on the behavioral biases tracked using ESIS strategy within the iterative querying process in AL environment. Further, it is noted that the bias-aware decision models suffer a loss in performance as compared to naive models during the initial query steps. This is due to the computational effort spent in tracking the behavioral biases and subsequently modifying the training weights of individual samples during the iterative querying process. However, as the behavioral biases are tracked using ESIS strategy and appropriate handling steps are taken, the bias-aware decision models outperform naive models and the performance is significantly improved at the end of querying procedure.

6.5 Summary

In this chapter, a systematic framework for modeling, tracking and adaptation of behavioral biases in a collaborative human-AI decision setup is presented. Firstly, the human-AI

interactive decision environment is modled in the form of a coupled dynamical system (using communication graphs) that evolves as more queries are made in the context of AL frameworks. A novel ESIS strategy is proposed for tracking behavioral biases within AL settings. Finally, a model adaptation framework is introduced to handle the tracked biases. This is executed by allocating suitable weights to the training samples during the iterative querying process. The proposed framework is evaluated by conducting experiments on real-world pancreatic cancer dataset. We consider five different learning scenarios: (i) learning for low-grade expert; (ii) transition from low-grade to medium-grade expert; (iii) learning for medium-grade expert; (iv) transition from medium-grade to high-grade expert; and (v) learning for high-grade expert. In each of these learning scenarios, the classification accuracy scores and extent of behavioral biases over different queries for both "naive" and "bias-aware" decision models are recorded and analyzed. Naive models do not track or incorporate knowledge of behavioral biases during the learning process. On the other hand, bias-aware models adapt based on the behavioral biases tracked using ESIS strategy. It is observed that bias-aware models improve the classification accuracy of naive decision models by upto 16%. The next chapter presents two case studies on addressing observational bias in decision models.

Chapter 7

Observational Bias in Decision Models – Case Studies

DNNs have demonstrated remarkable advancements across diverse domains in the recent years. These data-driven models, typically characterized as "black-box" models, often encounter failures attributable to various factors. A primary limitation of these NN-based decision models is their propensity to overfit, resulting in sub-optimal generalization performance when extrapolating beyond the confines of the available data, particularly in out-of-sample prediction tasks. The predictions generated by neural network models may exhibit physical inconsistencies, thereby reducing their interpretability. One approach to address this challenge involves incorporating prior domain knowledge, which encapsulates a high-level abstraction of the underlying natural phenomena. The "black-box" models can be made more robust by incorporating additional biases.

Observational data serve as a cornerstone in the recent advancements of ML and represent the fundamental input mode through which biases can be introduced into ML algorithms [116]. With a comprehensive dataset covering the input space of a learning task, ML methodologies have showcased remarkable efficacy in accurately interpolating data points, even in tasks characterized by high dimensionality. Particularly in the domain of physical systems, the proliferation of sensor networks enables the exploitation of diverse observations spanning

various levels of fidelity, facilitating the monitoring of complex phenomena across spatial and temporal scales. These observational data ideally encapsulate the underlying physical principles governing their generation, presenting an opportunity to embed such principles into ML models during their training phase, albeit weakly. However, for deeply parameterized ML models, substantial data volumes are typically required to reinforce these biases and yield predictions that adhere to specific symmetries and conservation laws. Consequently, a primary challenge arises concerning the considerable expense associated with data acquisition, particularly in fields such as physical and engineering sciences where observational data may necessitate costly experiments or extensive computational simulations.

This chapter presents two case studies, each demonstrating a way to incorporate observational biases within the learning frameworks. Firstly, a GNN-based framework is developed for identification of potential biomarkers for biological phenomena. This involves incorporating knowledge from an existing database to develop a graph that incorporates protein-protein interactions as edge connections and is used as input data to identify the potential biomarkers. Secondly, a foundational model-based AL framework is proposed for fault diagnosis of electrical motors, that utilizes less amount of labeled samples, which are most informative and harnesses a large amount of available unlabeled data by effectively combining AL and Contrastive Self Supervised Learning (SSL) techniques.

7.1 Case Study 1: Identifying Potential Biomarkers

In this section, a GNN-based framework is presented to identify potential biomarkers for examining the change in protein expression changes at different stiffness levels in Glioblastoma studies. This is an extension of our prior work on identification of potential biomarkers for Traumatic Brain Injury (TBI). TBI represents a significant global burden in terms of mortality and morbidity. In the United States, the Center for Disease Control and Prevention reported more than 220,000 TBI-related hospitalizations and 60,000 TBI-related deaths in 2021 alone. TBI is a multifaceted neurological condition characterized by initial mechanical injury followed by secondary pathophysiological processes including neuroinflammation,

excitotoxicity, oxidative stress, cellular apoptosis, and mitochondrial dysfunction. These secondary mechanisms collectively contribute to long-term motor and cognitive impairments [117–120]. Presently, TBI diagnosis primarily relies on neurological assessments such as the Glasgow Coma Scale and imaging modalities such as Computed Tomography (CT) and Magnetic Resonance Imaging (MRI). However, these methods are limited in their ability to swiftly and accurately detect, evaluate, and monitor the severity and specific components of brain injury, such as cell death, neuroinflammation, or blood-brain barrier (BBB) disruption [121].

This study conducts a thorough investigation of the biophysical alterations and changes in DNA/protein content observed in circulating extracellular vesicles (EVs) across various phases of TBI, including the acute, post-acute, and chronic stages. The protein content of EVs was analyzed using targeted western analysis targeting established TBI biomarkers, as well as an unbiased proteomics approach is demonstrated to identify potential novel biomarkers. Employing biostatistical and computational methodologies, which include graph-based ML algorithms and protein-protein interaction networks, this study aims to explore a diverse array of TBI biomarkers encapsulated within EVs. The proposed GNN-based framework is then extended to identify potential biomarkers for examining the change in protein expression changes at different stiffness levels in Glioblastoma studies.

7.1.1 Proposed Methodology

The proposed framework makes use of GNN as it allows to systematically incorporate multiple features such as neurological associations, protein-protein interactions, and time-dependent label-free quantification (LFQ) intensity values of each protein. By including these parameters, GNN provides a more comprehensive analysis that better captures the biological context of each identified protein to increase the likelihood of discovering a potential biomarker. The proposed workflow is illustrated in Figure 7.1. Firstly, a Protein-Protein Interaction (PPI) network was constructed using available databases from the STRING tool [122], considering the list of proteins in the proteomics data as nodes of the network. The

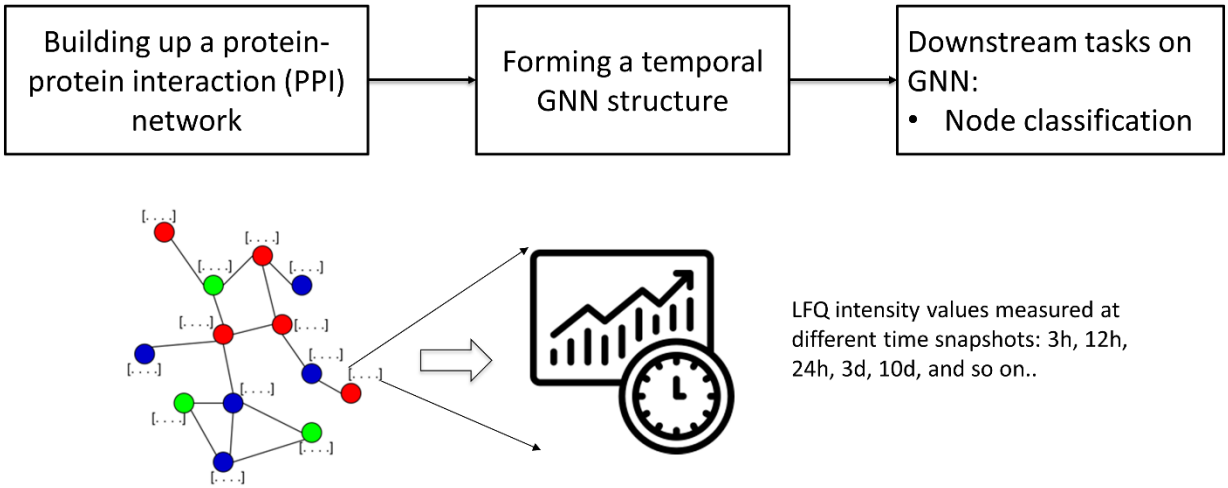


Figure 7.1: Proposed framework for identification of potential biomarkers.

interactions between individual protein molecules constitute the links/edges of PPI network. LFQ intensity values captured over different time stamps are pre-processed and attributed as node features. This helps to incorporate the temporal structure among LFQ intensity values of proteins within the GNN framework. Identifying potential biomarkers is formulated as a node classification problem, where the node labels represent the ranks (between 1-6) of corresponding protein being a potential biomarker. Rank 1 represents the highest probability, and Rank 6 represents the lowest probability of a protein being a potential biomarker. This flow is illustrated in Figure 1. A small fraction (¡ 5%) of nodes are labeled based on prior domain knowledge and inputs from existing databases. Once the graph is constructed, GNN-based learning techniques are implemented to predict the labels (i.e., probability of individual proteins being potential biomarkers) for all the nodes in the network. In this work, GraphSAGE (discussed in eq. (2.1)) is used as an inductive node embedding approach that concurrently learns both the topological structure and distribution of features for a node in its local neighborhood.

7.1.2 Results

For the case of TBI, the ranked list of potential biomarkers at different timestamps is presented in Table 7.1. A sample sub-networks demonstrating the topological relationships

Table 7.1: List of ranked potential biomarkers at different stages of TBI.

Rank	Time					
	3 hpi	**12 hpi**	**24 hpi**	**3 dpi**	**10 dpi**	**30 dpi**
1	Orm1 Apoa1 Saa1	Aldob Serpina1c Cir1 Apoc3	Orm1	Thbs1 Orm1 Serpina6	Orm2 Psma4	Thbs4
2	Apob Cir1 Hspa1b	Thbs1 Psma5 Serpina1a	Cir1	Apoa2	Apoa1 Orm1 Saa2 Rpl7	Hspa1a
3	Apoa2 C1sa Amy1 Aldoa Mup17	Orm2 Apoa2	Thbs4 Apoa2	Apoc2	Psmb8 Apoa2 Lcat	Apoa2 Serpina1b Psmb3
4	Thbs1	Orm1	Serpina1c Acta2 Psma4 Hspa1a	Amy1 Psma5 Psmb3	Thbs4 Psmb2 Serpina1b	Apoa4 Psma5 Orm1
5	Psma4	Apoa1 Psma2 F7 Rpl7	Cat Ces3a Serpina1d	Aldob Serpina1a Psmb6	Cir1 Aldh1a7	Apob
6	Prdx2	C1qa	Psma2	Serpina1e	Thbs1	Apoa1 Orm2 Serpina6

between potential biomarkers at all ranks as well as the non-potential biomarkers at 3 hours post injury (hpi) and 10 days post injury (dpi) timestamps are illustrated in Figure 7.2 and Figure 7.3 respectively. It is worthwhile to note that the potential biomarkers computed using the proposed methodology match with the ones identified via biostatistical analysis.

For the case of protein expression changes at different stiffness levels in Glioblastoma studies, the downstream task is formulated as node classification problem, where the node labels represent one out of high, medium or low. These labels indicate the probability of corresponding protein being a potential biomarker. The experiments are conducted at three different stiffness levels: 100 Pa, 4 kPa and 25 kPa. The potential biomarkers are evaluated for all the three possible stiffness transition scenarios, i.e., 100 Pa - 4 kPa, 4 kPa - 25 kPa,

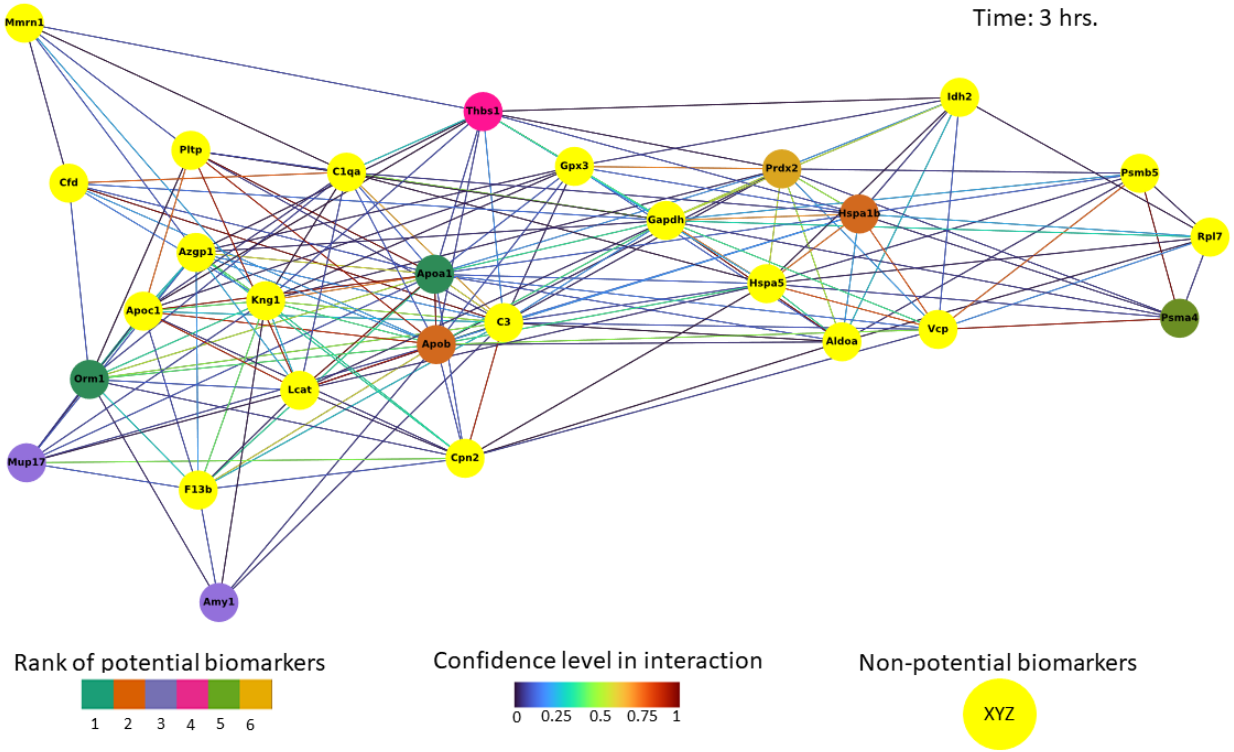


Figure 7.2: Protein-Protein network enrichment analysis of potential biomarkers at 3 hpi of TBI.

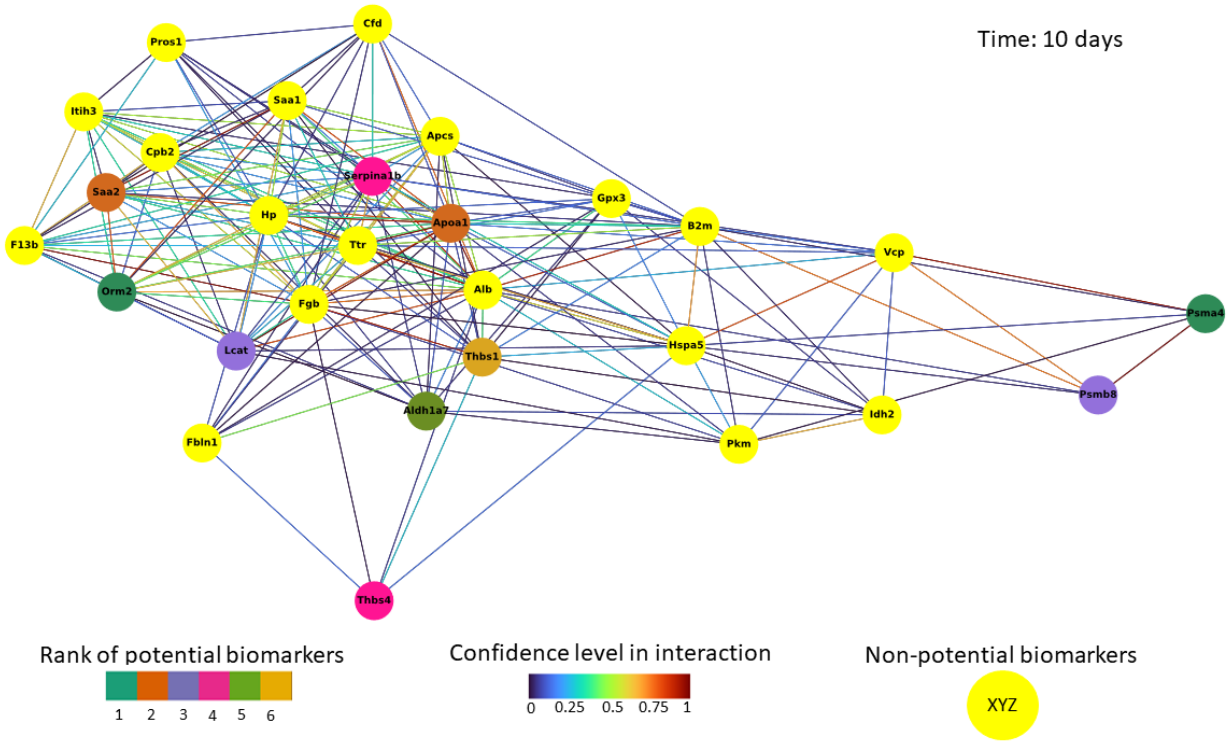


Figure 7.3: Protein-Protein network enrichment analysis of potential biomarkers at 10 dpi of TBI.

and 100 Pa - 25 kPa. The corresponding sample sub-networks demonstrating the topological relationships between potential biomarkers at different probability levels are illustrated in Figure 7.4, Figure 7.5, and Figure 7.6 respectively.

7.2 Case Study 2: Active Foundational Models

Electrical motors are the backbone of various industries, ensuring reliable and efficient operation across various applications, ranging from commercial and industrial settings to aerospace, computer systems, robotics, and defence. Despite their versatility and reliability, electrical motors are susceptible to various faults that can disrupt performance, jeopardize system safety, and lead to costly downtime. Timely and efficient fault detection in electrical motors is paramount to maintaining optimal functionality, reducing maintenance expenses, and preventing catastrophic failures.

Contemporary approaches for fault diagnosis of electrical motors encompass analytical

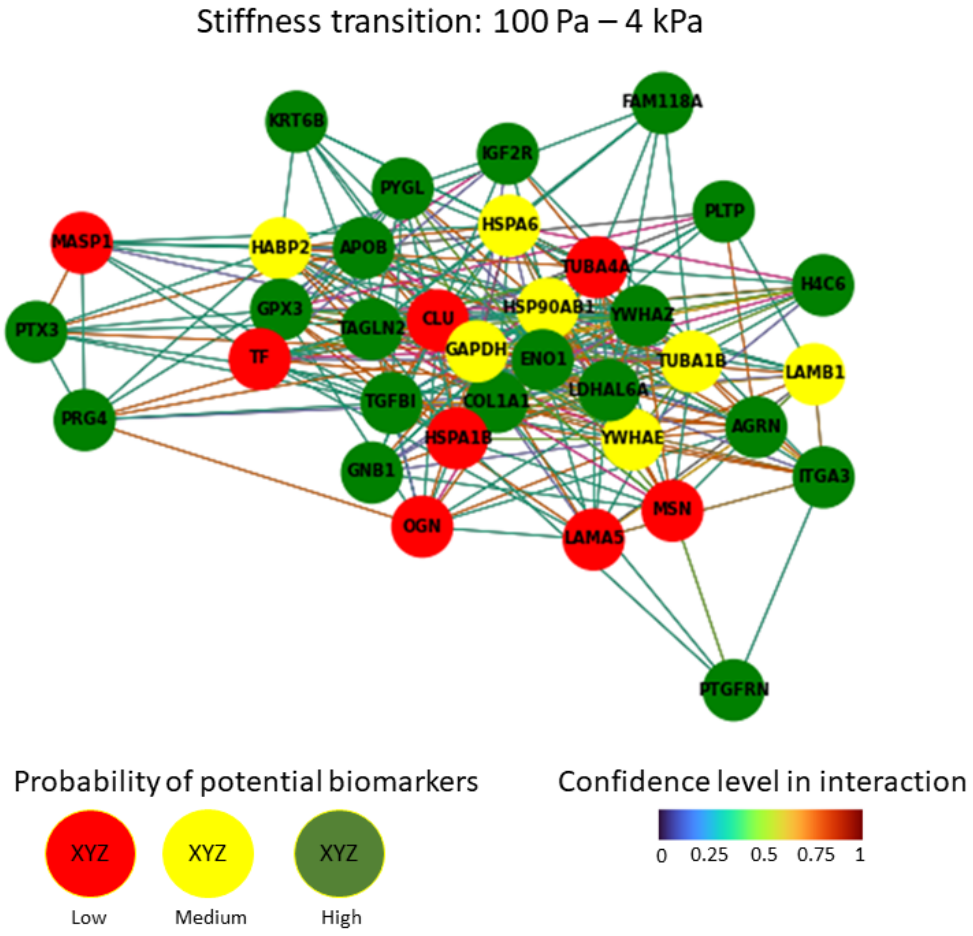


Figure 7.4: Protein-Protein network enrichment analysis of potential biomarkers for 100 Pa - 4 kPa stiffness transition in Glioblastoma experiments.

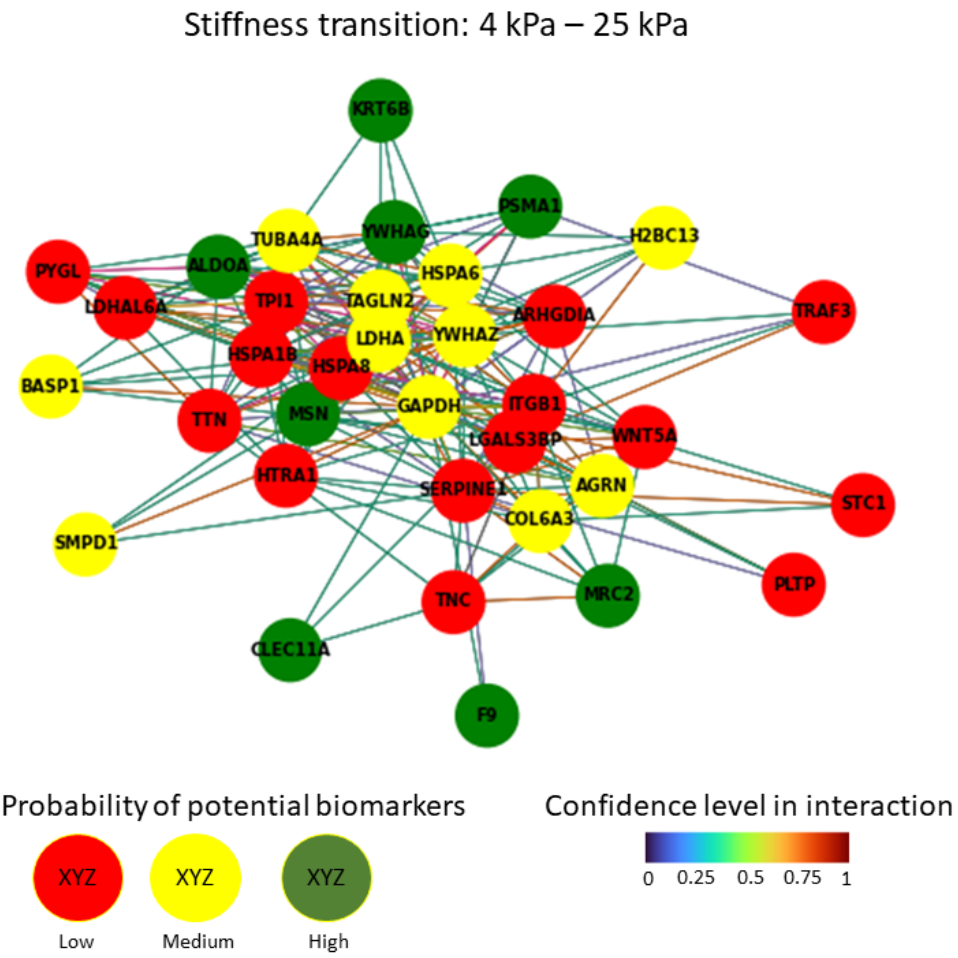


Figure 7.5: Protein-Protein network enrichment analysis of potential biomarkers for 4 kPa - 25 kPa stiffness transition in Glioblastoma experiments.

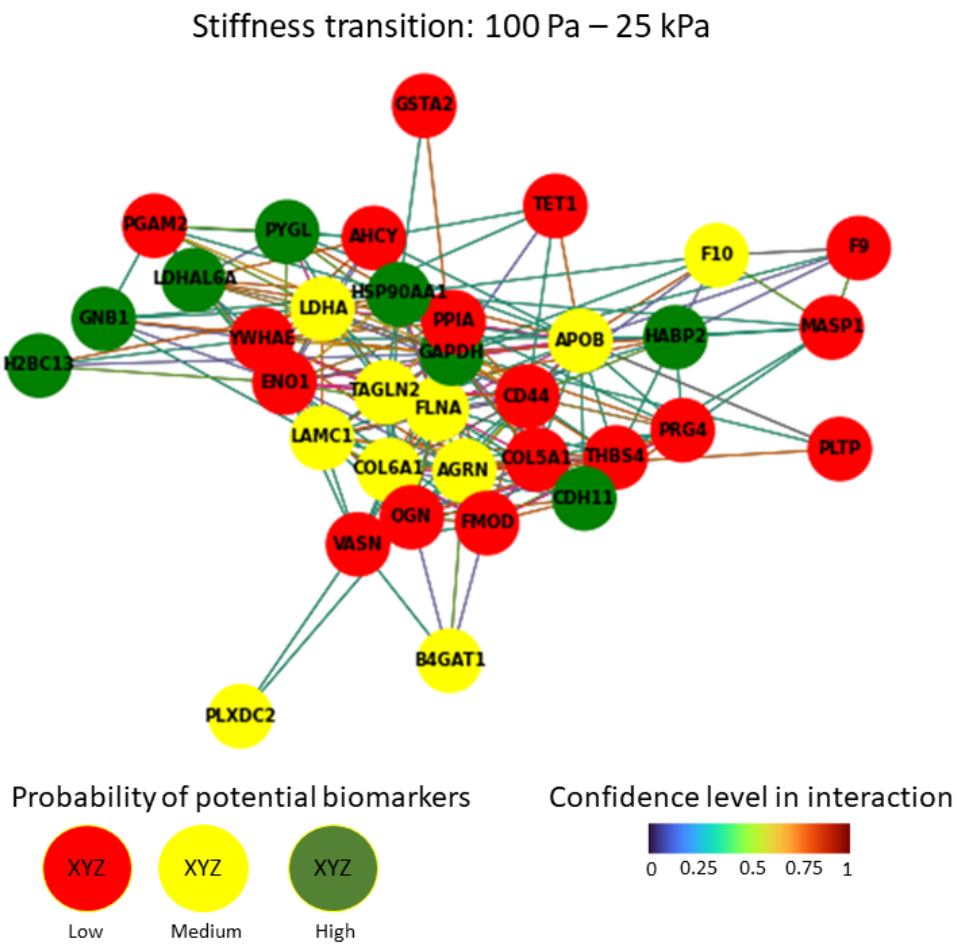


Figure 7.6: Protein-Protein network enrichment analysis of potential biomarkers for 100 Pa - 25 kPa stiffness transition in Glioblastoma experiments.

models and data-driven methodologies. The analytical and physics-based model relies on signal processing techniques to extract domain-invariant features shared among various faults, thereby developing the motor fault diagnosis model [71, 123–125]. Nonetheless, a limitation of this approach arises from the challenge of identifying universal physical features applicable to all fault types. In contrast, data-driven models leverage ML and DL techniques, often requiring less expertise than their analytical counterparts. Traditional ML methods, including Artificial Neural Networks (ANN), Support Vector Machines (SVM) [126], Random Forest (RF) [127], and kNN [128], are widely used for fault detection. These ML approaches come into play following signal processing to extract features from raw vibration data. Notably, these ML methods are considered shallow networks and may encounter difficulties in effectively extracting fault-related features from raw vibration data in complex environments.

DL constitutes a pivotal branch of ML, distinguished by its superior implicit feature extraction compared to shallower machine learning methods. The efficacy of deep learning is particularly notable as the quantity of labeled data scales up. Within the realm of intelligent fault diagnosis, various deep learning methods, such as Convolutional Neural Networks (CNN), Recurrent Neural Networks (RNN), and Long Short-Term Memory (LSTM), have recently emerged as focal points of research [129–131]. One notable development in deep learning is the Transformer model, a technique centring on attention mechanisms (AM) and widely employed in diverse research domains [132]. The uniqueness of the transformer lies in its capacity to assign greater importance to crucial features while extracting domain-discriminative features from input data, all through an attention mechanism. This innovative model is founded on a novel encoder-decoder architecture, primarily relying on the multi-head self-attention mechanism and feed-forward neural networks. It is designed to tackle sequence-to-sequence tasks with an increased ability to capture global dependencies [133]. In contrast to alternative networks like convolutional and recurrent networks, Transformer-based models demonstrate superior performance in various computer vision applications [134]. Recent work in [133] introduced a fault diagnosis approach that leverages a transformer with an optimized stacked autoencoder for analysing rotating machinery.

However, even with these advancements, many deep learning algorithms for motor fault

diagnosis often rely heavily on abundant labeled samples and the assumption that training and testing data originate from the same distribution, which is not valid in practical scenarios. Acquiring fault data under real-world conditions can be challenging and may involve extended machine operation under fault conditions, which is often impractical. Consequently, gathering an ample number of labeled samples can be problematic. Deep learning models can struggle to establish effective decision boundaries when labelled samples are scarce, resulting in suboptimal performance. Despite collecting substantial amounts of motor condition monitoring data, manual annotation remains a costly, error-prone, and labor-intensive process. This frequently results in a significant portion of data remaining underutilized due to either the absence of meaningful labels or insufficient samples for specific fault cases, particularly in scenarios with limited labeled data. In such circumstances, supervised learning approaches often perform inadequately. Existing semi-supervised methods struggle to address the challenge of fault detection with extremely limited labeled samples [135, 136].

Self-supervised learning can effectively overcome this limitation, efficiently extracting meaningful features from raw, unlabeled data, thus eliminating the need for expensive labels [35]. Currently, numerous self-supervised learning methods have been developed, primarily in the domain of image analysis. These methods involve enabling the network to learn visual features by constructing pretext tasks, such as predicting image rotation, image coloring, random cropping, and more. However, mechanical vibration signals differ significantly from image data. They are characterized by non-stationary patterns and strong periodicity, necessitating unique pretext tasks tailored to these distinctive properties. Popular signal transformation techniques are commonly employed to define pretext tasks, including random zeroing, random increase, random decrease, time disorder, and random jittering [137]. Encoders trained to predict these pretext tasks are expected to learn general features that prove valuable for downstream tasks that typically require expensive annotations. Another prominent category of self-supervised learning techniques utilizes contrastive losses across various computer vision applications [138]. These methods aim to create a latent space that brings positive samples closer together (e.g., adjacent segments from the same time series signal) while pushing apart negative samples (e.g., segments from different time se-

ries signals). Recently, a variant of contrastive learning known as instance discrimination has exhibited remarkable performance in various downstream tasks [35]. In this instance discrimination paradigm, the latest work has introduced the concept of non-trivial positives among augmented samples from the same image and even across different images [139]. Instead of focusing on clustering or prototype learning, they maintain a support set of image embeddings and employ nearest neighbors from this set to define positive samples.

Taking inspiration from these advancements, an instance discriminative technique is introduced with the nearest neighbor from the support set as a positive sample to the field of electrical motor fault diagnosis for the first time. This pioneering approach has demonstrated a significant improvement over existing self-supervised techniques, marking a promising advancement in this domain. In all the tasks related to fault classification, obtaining labels for data is often laborious and expensive, while a vast pool of unlabeled data remains readily accessible. Furthermore, training datasets frequently contain redundant samples, leading to protracted classifier training without commensurate improvements in classification performance. To address these challenges, the development of Active Learning (AL) techniques has aimed to pinpoint the most valuable samples for manual labeling, ultimately streamlining the classifier training process [140]. AL has emerged as a pivotal concept, promising to reduce annotation costs. It typically encompasses iterative procedures in which the learning algorithm gains autonomy in selecting additional training data. One of the pupular query stratgies in AL is uncertainty sampling, which is designed to solicit labels for samples characterized by high uncertainty. This uncertainty can be quantified through posterior probabilities and marginal entropy, as various strategies for identifying uncertain samples are explored in the literature [141]. Subsequently, these selected informative samples are manually labeled and seamlessly integrated into the training process, enhancing the classifier's performance. While AL is recognized as a supervised approach incurring higher annotation costs than self-supervised learning (SSL), this study critically evaluates the advantages and limitations of both AL and SSL methodologies. As a result, an innovative fault diagnosis approach is introduced for industrial equipment that harmoniously integrates SSL with AL.

The key contributions of this work are as follows:

1. A novel foundational model-based AL framework is introduced for electrical motor fault diagnosis. This model consists of two key components: the construction of the backbone model, followed by the fine-tuning procedure for various target tasks using less amount of labeled samples.

2. Instead of solely relying on data augmentation techniques, we have employed an advanced method called instance discrimination-based contrastive SSL to train the transformer-based backbone model. In contrastive SSL, we have harnessed the benefits of using the nearest neighbor as a positive sample. This approach allows the transformer encoder to learn a more robust representation of the vibration signal, resulting in a well-generalized backbone model.

3. AL strategy is implemented based on entropy and Kullback-Leibler (KL) divergence as combined heuristics for strategic sample selection. This helps in identification of most informative samples for annotation within the iterative training loop. The experimental results demonstrate that the proposed model can learn effective decision boundaries even with a limited number of labeled samples. This achievement overcomes the laborious and costly task of manual labeling.

4. The strengths of AL and the nearest neighbor contrastive SSL methods are amalgamated. A framework is proposed for constructing the backbone model, enabling it to learn more refined representations of the data with minimum labeled samples. Consequently, the proposed model is highly effective in various downstream tasks within the same machine as well as across different machines. The proposed approach has been thoroughly evaluated using three publicly available bearing fault datasets.

7.2.1 Proposed Methodology

In this work, an active foundational model is proposed for fault detection and diagnosis that can be employed for target tasks across different electrical motors. This is achieved by effectively combining the aspects of AL and contrastive SSL. The proposed model $\mathcal{M}$

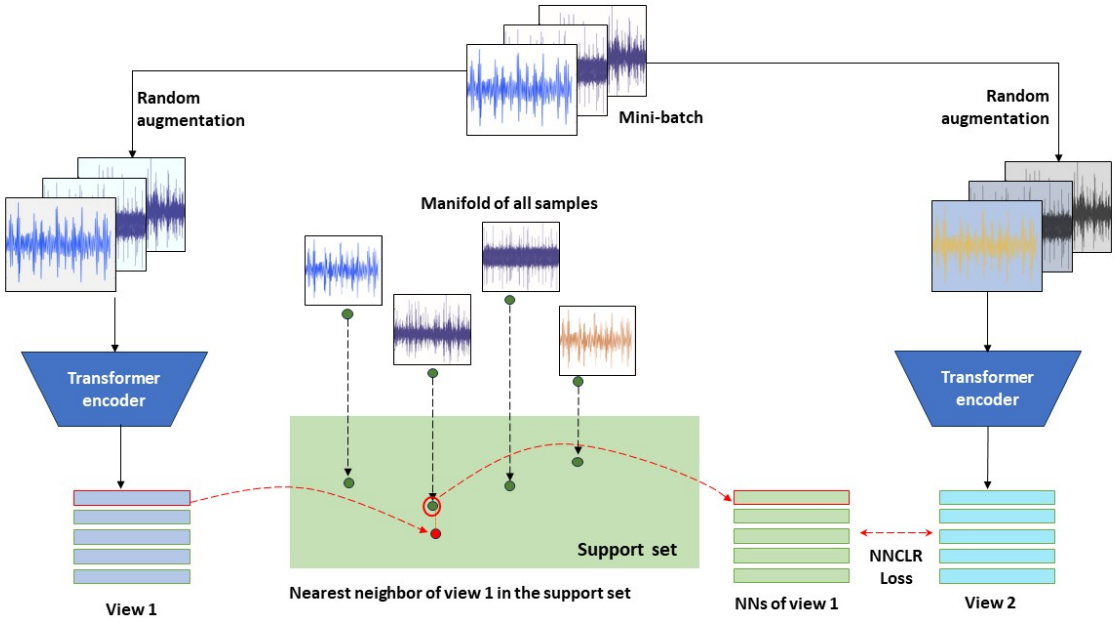


Figure 7.7: NNCLR training for developing active foundational models for fault diagnosis of electrical motors.

consists of two components, a backbone encoder $\mathcal{B}$ and a feed-forward neural network $\mathcal{N}$. The backbone training framework is primarily inspired by Nearest Neighbor Contrastive Learning Representation (NNCLR) framework [139], where contrastive Learning framework is utilized, as discussed in Section 2.6, with slight modifications in terms of robustness to a wide range of semantic augmentations. The implementation of the NNCLR framework in the context of 1-dimensional vibration signal input is schematically illustrated in Figure 7.7. This framework replaces one of its augmentations with its nearest neighbor from the dynamic feature space set $\mathcal{S}$ during the forward pass of the training loop. The adoption of nearest neighbors as positive samples in the contrastive loss function results in a modified loss function as shown below:

$$l(i,j) = -\log \frac{\exp\left(\text{sim}\left(\text{NN}(z_i, \mathcal{S}), z_j\right)/\tau\right)}{\sum_{k=1}^{n} \mathbb{I}_{[k \neq i]} \exp(\text{sim}\left(\text{NN}(z_i, \mathcal{S}), z_k\right)/\tau)} \tag{7.1}$$

where,

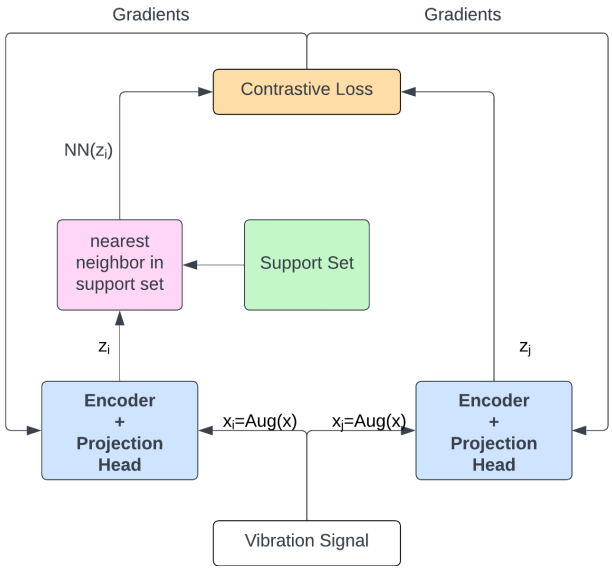
$$\mathrm{NN}(z, \mathcal{S}) = \arg\min_{s \in \mathcal{S}} \|z - s\| \tag{7.2}$$

The objective of the Contrastive Learning framework is to learn the representation space where "similar" samples are closer. The training of Backbone $\mathcal{B}$ and Feed Forward Network $\mathcal{N}$ takes place separately where the $\mathcal{B}$ learns the representation space of similar samples and the $\mathcal{N}$ draws better decision boundaries. The flowchart of the proposed training method is demonstrated in Figure 7.8. Once $\mathcal{B}$ and $\mathcal{N}$ are trained appropriately, their performance is improved by iteratively querying the most informative samples from the unlabeled pool of data by AL within a semi-supervised setting. Entropy of class probabilities and KL divergence [142] between actual probability distribution to be predicted and the predicted probability distribution by $\mathcal{M}$ for the particular sample are used as AL heuristics for the semi-supervised training.

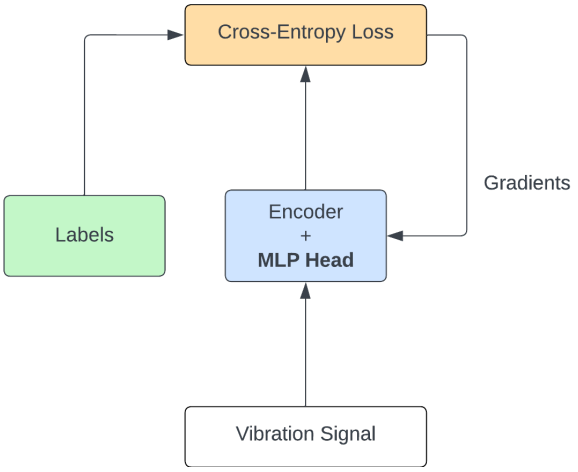
NNCLR framework

Training $\mathcal{B}$ and $\mathcal{N}$ in the NNCLR framework according to Figure 7.8 requires the following terminology to be defined:

- *Contrastive Augmenter:* A signal augmentation module that employs robust data augmentations on the input signals, guided by the hyper-parameters specified for contrastive augmentation.
- *Classification Augmenter:* A signal augmentation module that utilizes mild data augmentations on the input signals, following the hyper-parameters designated for classification augmentation.
- *Transformer Encoder:* An encoder network based on the Transformer architecture, tasked with extracting feature embedding of a high-dimensional nature from the augmented signals.
- *Projection Head:* A non-linear MLP that maps the feature embeddings to a lower dimensional space.



(a) Backbone Training in NNCLR Framework



(b) Feed Forward Network Training in NNCLR Framework

Figure 7.8: Training flowcharts of $\mathcal{B}$ and $\mathcal{N}$. Bold indicates that Gradients are not frozen and BackProp is enabled

- *Linear Probe:* A single dense layer that maps the feature embedding to the logits for the number of output classes specific to the dataset being trained.

In addition to the aforementioned SSL training blocks in the architecture, NNCLR [139] demonstrates the advantage of using nearest neighbors as positive samples in contrastive self-supervised learning. This approach is used because it has been proved to be superior to an augmented view embedding method like SimCLR [35] for contrastive learning. Data enrichment is essential for contrastive self-supervised learning systems to operate optimally. However, NNCLR is less reliant on intricate data augmentation methods because the samples from the closest neighbors already have a wide range of variations. Therefore, data augmentation is employed along with the nearest neighbors as positive samples. The steps involved in data augmentation and obtaining the nearest neighbors are discussed as follows:

- *Data Augmentation:* In this work, two techniques for data augmentation, i.e., random jittering and random zeros are used. Random jittering is achieved by introducing random noise to the time series data. Typically, this noise is taken from a random distribution with a known mean and variance. The variety and uncertainty in machine data operating in varying circumstances can be simulated by adding random noise. Random zeros involve randomly adding zeros to time series data, which entails randomly zeroing out parts or segments. This data augmentation method is beneficial for simulating real-world situations where some information may be missing or unreliable by performing data dropout. A probability or percentage that expresses the chance of zeroing out a data point or segment can be used to start the random zero procedure. The frequency of missing data points is controlled by a probability parameter.

- *Support set:* In order to get the nearest neighbors, NNCLR framework utilizes a support set that stores the embedding of a subset of the dataset in the memory. The support set is continuously replaced during the training. Increasing the size of the support set helps in improving overall performance. The likelihood of approaching the nearest neighbor from the entire dataset improves when an extensive support set is used. In contrastive losses, the nearest neighbors in the queue are employed to locate positives.

Table 7.2: Hyper-parameters of the transformer encoder employed in the proposed active foundational model.

Parameters	**Value**
Input time series size	192×1
Position encoding format	1D
Number of stacked transformer blocks in the Transformer layer	4
Number of heads in MSA	4
Hidden layer size in a feed-forward network inside transformer	4
Embedding dimension of the nonlinear transformation in MLP	256
Dropout probability	0
Optimizer	Adam
Initial Learning rate of the optimizer	0.001

Random augmentations, like random increases, decreases, or zeros, cannot provide positive pairs for various views, deformations of the same signal, or even other identical instances within a semantic class. The data augmentation pipeline is heavily responsible for generalization but cannot account for all variations within a particular class. NNCLR proposes to move beyond single instance positives. This allows to develop features more resistant to various views, deformations, and variations within a single class.

Implementation Details

A time series transformer network serves as the encoder in the training setup. The transformer, which is state-of-the-art and used in LLM (Large Language Model) and computer vision applications, is highly scalable, transferrable, and generalizable [143]. Table 7.2 lists the proposed transformer encoder hyperparameter specifications in detail. The architecture is then supplemented with the projection MLP head. It features two fully connected layers and an input layer. Batch-normalization is applied to every fully connected layer, and the ReLU activation function is applied to every layer after batch-normalization, except the last layer. The next block in the architecture has prediction MLP. It has two fully connected layers. Batch-norm and ReLU are placed after the hidden layer of prediction MLP. The output layer, the final layer in the prediction MLP, has nodes equal to the number of output classes with Softmax activation.

Backbone training

The NNCLR framework-based transformer encoder network is treated as a backbone network $\mathcal{B}$. It is trained with a dataset consisting of various mechanical faults and a healthy state of machines in both constant speed and varying speed conditions. A recently published experimental dataset [144] is used for training $\mathcal{B}$. Healthy state, bearing fault (inner and outer race faults) and shaft misalignment faults from the dataset are considered to train the backbone model $\mathcal{B}$. In this work, for the first time, an active learning approach with contrastive learning is introduced to improve the model performance by enabling it to learn from the most informative samples and letting the model learn the precise decision boundaries. Entropy is used as an AL heuristic to determine informativeness within the unlabeled samples as it is the most popular option for uncertainty-based querying approaches. This is because it easily generalizes to probabilistic multi-class annotations and models more complex structural data. Initially, $\mathcal{B}$ is trained with 15% of the labeled samples. Then, the samples are queried within an AL setup. The Shannon entropy of the predicted probability distribution over the classes for all the given samples is calculated. Based on this, say $2K$ samples are queried that have the highest entropy, labeled and KL-Divergence is calculated as follows:

$$D_{KL}\left(q_{\mathcal{L}}^{x}||p_{\mathcal{M}}^{x}\right) = \sum_{c} q_{\mathcal{L}}^{x}(c) \cdot \log \frac{q_{\mathcal{L}}^{x}(c)}{p_{\mathcal{M}}^{x}(c)} \tag{7.3}$$

where, $p_{\mathcal{M}}^{x}$ is the predicted probability distribution for sample x by model $\mathcal{M}$ over the classes and $q_{\mathcal{L}}^{x}$ is the actual probability distribution to be given for the sample x given its label class $\mathcal{L}$. It is defined as follows,

$$q_{\mathcal{L}}^{x}(c) = \begin{cases} 1 & \text{if c is } \mathcal{L} \\ 0 & \text{else} \end{cases} \tag{7.4}$$

Re-training the model with samples having highest KL-Divergence shall help improve both accuracy as well as confidence. In this manner,an additional 5% of the unlabeled data are annotated using AL and the model is subsequently re-trained. It is observed that the proposed training strategy enables achieving a classification accuracy of 94% with just

20% labeled samples. On the other hand, to obtain that higher prediction accuracy with the random annotation method, it requires more than 60% labeled samples, which clearly demonstrates the efficacy of the proposed approach.

Target Tasks and Fine-tuning

The target tasks considered in this work are divided into two groups to evaluate the performance of the foundational model $\mathcal{M}$ across fault cases and operating conditions (a) within the same machine, and (b) different machines. A hierarchical approach is followed to design the tasks in the first group. The initial few tasks aim to categorize the samples into healthy vs. faulty. Once the fault is detected, the subsequent target tasks are designed to diagnose the type, location, and severity of faults. Such a hierarchical approach enables a comprehensive fault diagnosis, offering detailed insights into the nature and location of potential faults. This is applied to both constant speed and varying speed conditions. The list of target tasks in the first group is tabulated in Table 7.3. The second group of target tasks is designed to showcase the scalability and transferability of the proposed model $\mathcal{M}$. The experimental vibration data from three different machines are considered to evaluate our model performances across machines. The transformer encoder from learning the new data representation was frozen, and just the projection MLP head $\mathcal{N}$ was made trainable for performing downstream target tasks designed with the datsets for new machines. Only a small amount of labeled samples was used for fine-tuning the MLP head $\mathcal{N}$ for various downstream tasks, whereas the backbone network $\mathcal{B}$ was frozen with the weights that have been learned for the base machine dataset [144]. The output layer of MLP head $\mathcal{N}$ was adjusted to the number of outputs based on the number of classes in the target tasks.

7.2.2 Results

The backbone model is trained using the data collected from a bearing test rig at the Korea Advanced Institute of Science and Technology (KAIST) [144]. In order to verify and analyze the effectiveness of the proposed model in fault diagnosis across different machines, datasets corresponding to three different machines are utilized: CWRU bearing dataset [70],

Table 7.3: Target Tasks performed with active foundational model

S. No.	Target Tasks
1	Fault Detection at constant speed: Classification - Healthy vs. Faulty
2	Fault Diagnosis at constant speed: Classification - Bearing Fault vs Shaft Misalignment vs Rotor Imbalance
3	Bearing Fault Diagnosis at constant speed: Classification - Inner Race vs. Outer Race
4	Fault Diagnosis at variable fault size: Classification - Healthy vs. Bearing Fault vs Shaft Misalignment vs Rotor Imbalance
5	Bearing Fault Diagnosis at variable fault size: Classification - Inner Race vs Outer Race

University of Ottawa dataset [145], and HUST bearing dataset [146]. In all the experiments, results from our prior work in [38] are considered as baseline.

Efficiency

Using the proposed approach,$\mathcal{B}$ is trained efficiently with less number of labeled samples. The contrastive learning method enables the transformer encoder to learn a better representation of the multiple fault cases from the KAIST dataset. The backbone $\mathcal{B}$ is trained with 15% of labeled samples and attained 89% classification accuracy. A better performance was observed after incorporating the AL query strategy to select the most informative samples and subsequently re-train the backbone using the proposed approach. When random sample selection method was implemented for training $\mathcal{B}$, it took 60% labeled samples to reach 94% classification accuracy. However, when the AL querying strategy is employed to select samples from an unlabeled pool of samples, 94% accuracy was attained with just 20% labels. To evaluate the effectiveness of the trained backbone network $\mathcal{B}$, a list of hierarchical target tasks is developed within the same machine data, and it is demonstrate that the proposed model $\mathcal{M}$ can perform better than baseline [38] in most of the target tasks presented in Table 7.4. The list of hierarchical target tasks framed from the KAIST dataset is shown in Table 7.3. The performance of the $\mathcal{B}$ in the hierarchical target tasks is shown in Table 7.5. The results depict that the performance of the backbone $\mathcal{B}$ is higher due to the addition of the NNCLR framework with a transformer encoder network.

Table 7.4: Comparing performance of proposed backbone model vs. baseline backbone model for target tasks within the same machine.

Target Task	Classification accuracy with 25% labels	
	Proposed Model	Baseline Model
1	97.56	95.33
2	99.62	99.46
3	95.29	99.33
4	95.07	93.74
5	99.68	99.39

Table 7.5: Fine-tuning the backbone for different target tasks within the same machine

Target Tasks	Percentage of data used for fine-tuning				
	5%	10%	15%	20%	25%
1	92.06	94.80	95.61	96.29	97.56
2	96.30	97.84	98.80	99.20	99.62
3	90.90	92.88	93.82	94.85	95.29
4	82.07	89.85	92.53	94.54	95.07
5	97.43	97.73	98.46	99.24	99.68

Generalization

Generalization, or the adaptability of a model, refers to its capacity to grasp and extrapolate from intricate combinations of more minor elements or attributes. This capacity is crucial for enabling the successful application of the model to novel contexts and environments, particularly in the context of fault diagnosis. The extent to which the model can grasp the interactions among diverse factors or components and discern their influence on the overall observed state or condition is the key factor that determines its level of generalizability. This, in turn, empowers the model to apply its knowledge effectively and make accurate predictions in unfamiliar and diverse settings. A generalized model can effectively handle variations in input and adapt to various environmental conditions. To express the generalization ability of the proposed model $\mathcal{M}$, several downstream tasks were performed in datsets corresponding to three different machines. For all downstream tasks, the backbone model was fine-tuned by allowing only the MLP head to be trained while keeping the remainder of the encoder

Table 7.6: Generalization: Fine-tuning the backbone for different target tasks in different machine (CWRU dataset)

Loading condition	Data used for fine tuning		
	5%	10%	15%
0 HP	94.70	97.62	98.76
1 HP	82.46	86.79	89.42
2 HP	82.14	87.11	90.89
3 HP	88.64	92.54	93.97

Table 7.7: Generalization: Fine-tuning the backbone for different target tasks in different machines (HUST dataset with two different types of bearings)

Loading condition	Percentage of data used for fine-tuning					
	5%		10%		15%	
	Type 1	Type 2	Type 1	Type 2	Type 1	Type 2
0 W	82.10	89.20	87.28	92.15	88.92	93.17
200 W	82.25	89.06	87.00	91.95	90.10	93.79
400 W	80.27	85.77	85.58	89.48	87.87	92.56

components frozen with the initial weights used during the training of the $\mathcal{B}$. The experimental results demonstrating the generalization ability of the proposed modeling approach across CWRU and HUST datasets are presented in Table 7.6 and Table 7.7, respectively. It can be observed that the predictive performance improves as more amount of data is used for fine-tuning the decision models. This is intuitive because a better representation is learned as the amount of input training data is increased.

7.3 Summary

In this chapter, two case studies are presented to demonstrate ways of incorporating observational biases within the learning frameworks. The first case study illustrates a GNN-based framework for identification of potential biomarkers for biological phenomena. It involves incorporating knowledge from an existing database to develop a graph that incorporates protein-protein interactions as edge connections and is used as input data to identify the po-

tential biomarkers. This process entails leveraging information from an existing database to construct a graph wherein protein-protein interactions are represented as edge connections. This graph serves as input data for the identification of potential biomarkers. The second case study proposes a novel active foundational model for fault diagnosis of the electrical machines by combining aspects of AL and contrastive SSL during backbone training. The proposed framework consists of two major components: (i) constructing the backbone fault diagnosis model, and (ii) fine-tuning based on specific requirements of various target tasks. An initial model is trained using discrimination-based contrastive SSL technique, followed by re-training the same using informative samples selected via AL heuristics, namely, entropy and KL-Divergence. Once the backbone is trained, it is fine-tuned based on the requirements of different target tasks by updating the MLP projection head. The experimental evaluation has demonstrated the efficiency and generalizability of the proposed approach by achieving superior classification accuracy across several target tasks within the same machine as well as across different machines. The generalization capabilities of the proposed modeling approach makes it a valuable tool for real-world applications in fault diagnosis for electrical motors. The next chapter delves into explainability of decision models by conducting two different case studies.

Chapter 8

Explainability of Decision Models – Case Studies

Despite the fact that AI-powered systems have provided competitive benefits in the recent years, the black-box nature prohibits the explainability of their decisions and drives them to lack transparency. This issue prompted the development of explainable artificial intelligence (XAI), which supports AI systems that can explain their internal processes and decision-making methods [7].

This chapter presents two case studies to demonstrate different ways of explaining the predictions made by decision models. The soft hierarchical decision structure proposed for early-stage detection of pancreatic cancer provides confidences associated with the predicted labels in the form of probability values for each class. The differences between these probability values provide an indication of confidence associated with the corresponding prediction. It helps the doctors to determine whether additional tests are required for proper diagnosis. The active explainable decision framework proposed for early-stage detection of ovarian cancer is supported by feature relevance explanation technique. In this framework, the feature relevance scores explain as to how much each biomarker is contributing to arrive at a prediction for the sample under consideration.

8.1 Case Study 1: Early-stage Detection of Pancreatic Cancer

Pancreatic Cancer (PC) is the third leading cause of cancer-related death in the US with a 5-year survival rate of 11% [147]. PC is characterized by a poor prognosis, invasiveness, rapid progression, and profound resistance to drug treatment, all which results in poor outcomes [148], [149]. Because of the virtual absence of early warning signs, PC is infrequently diagnosed at an early-stage [150], [103]. Consequently, neither surgical treatment, nor chemo- and radiotherapy are effective against PC [148, 149, 151, 152]. Currently, no universal screening tests for pancreatic cancer exists, and the best techniques available for pancreatic cancer detection are the commonly used ones, which include, biopsy and imaging test like endoscopic ultrasound, CT scans, MRI, and Positron Emission Tomography (PET) scans [153]. The medium survival rate of PC drops sharply with a later stage of detection. According to the American Cancer Society, the 5-year relative survival rate for PC is 39% at the localized state, 13% at the regional state, and 3% at the distant stage [147]. Therefore, a feasible and cost-effective Liquid Biopsy [154] for PC detection would be of great value, if it is capable of detecting PC at the localized stage, preferentially by means of a simple blood test. Liquid biopsies are of big interest for diseases like pancreatic cancer, where tissue samples are limited. Some liquid biopsies exploited for PC consist of identifying and quantifying tumor-associated components released from all tumor sources that can be present in blood, serum, or plasma, such as circulating tumor DNA (ctDNA), circulating tumor cells (CTCs), and extracellular vesicles (EVs) [155, 156]. Major difficulties with these liquid biopsy technologies include the lack in ability to isolate pure tumor-associated components, which typically contains a mix of tumor- and non-tumor associated components, which makes this technology insufficient to stand alone [156]. However, with the exception of the protease-activity technology discussed here, none of the "classic" approaches to Liquid Biopsies, such as the capture and detection of circulating tumor cell or circulating tumor DNA, DNA-methylation studies or the analysis of the content of extracellular vesicles, are

capable of reliable detecting stage 1 PC [157–161].

The Bossmann group has established a panel of seven proteases (caspases B and E, matrix metalloproteinases (MMPs) 1, 3, and 9, urokinase plasminogen activator (UpA), and neutrophil elastase) and arginase as suitable panel of enzymes for early PC detection in 2018 [103]. This selection was based on Gene expression analysis [162] using data from NCBI GEO, Entrez Gene ID, Unigene ID and Gene Symbol [162]. Protease and Arginase activities in serum were measured with Fe/Fe_3O_4 core/shell nanobiosensors with an average particle size of 15 nm [103], [163], [164], [165], [166], [1]. The function principle is shown in Figure 8.1 (A). Each protease cleaves its respective consensus sequence and releases the fluorophore TCPP, which escapes the Förster quenching sphere of the nanoparticle plus tethered cyanine 5.5 dye (FRET pair) [163]. Upon escape, TCPP fluorescence increases and can be detected by a clinical plate reader. The fluorescence signal correlates with the fluorescence intensity [103]. Note that the nanobiosensor for arginase activity detection is not cleaved. Arginase performs a "post-translational" modification converting peptide-bound arginine into ornithine. The latter changes the dynamic of the peptide tether, which increases TCPP fluorescence [165].

Although statistically significant differences of the protease/arginase activity pattern of the group of all pancreatic cancer patients ($n = 31$) and the group of healthy, age- and gender-matched volunteers ($n = 48$) could be established utilizing these Fe/Fe_3O_4-based nanobiosensors, the overall sizes of the investigated groups was too small to establish the feasibility of early PC detection beyond reasonable doubt. Furthermore, simple biostatistics (e.g. performing Welch tests [167] and calculating p-values between data groups [167], [168]) do not provide the maximal extractable information from ultra-sensitive protease/arginase analyses. Therefore, information fusion based hierarchical decision structures are employed in this work for early-stage detection of pancreatic cancer. The classification models based on hierarchical decision structures are attracting significant research attention in the recent years. This is because they have demonstrated an appreciable predictive performance on a wide variety of interesting engineering applications like text classification [169], intrusion detection [170], manufacturing [171] and credit scoring prediction [172]. Biomedical applications like generation of molecular graphs [173], lung nodule malignancy classification

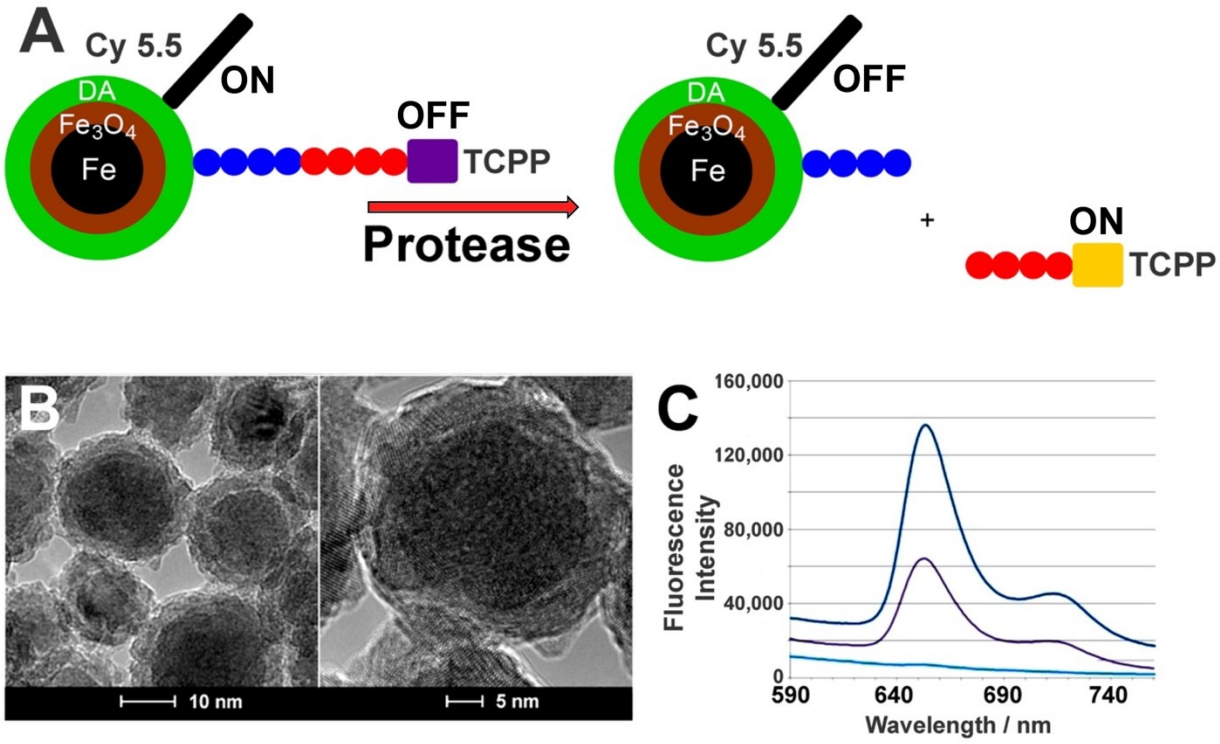


Figure 8.1: (A): Design principles of a nanobiosensor for protease detection. The OFF mode occurs when distance between fluorophore TCPP (tetrakis-carboxyphenyl-porphyrin), Fe/Fe_3O_4 nanoparticle, and FRET-acceptor cyanine (Cy) 5.5C is reduced, upon cleavage of the oligopeptide tether by a suitable protease present, this distance increases and leads to an increase in fluorescence intensity, which is called the ON mode. (B): TEM and HRTEM of dopamine-coated Fe/Fe_3O_4 core/shell nanoparticles. (C): Typical emission spectra occurring from a nanosensor for protease detection after 1h of incubation at 37 ^{0}C ($\lambda_{exc} = 421$ nm). low: buffer; middle: nanosensor; high: nanosensor after incubation with the respective enzyme; with permission from reference [1], copyright Elsevier, Amsterdam 2021.

[174], COVID-19 detection [175], skin lesion classification [176] and detection of Alzheimer's disease [177] have also incorporated the use of hierarchical learning methods to build efficient classifiers. However, such hierarchical decision models have not been proposed for the early-stage detection of PC. Given the limited sample size in such studies, exploiting a hierarchical classification structure helps to reduce the complexity of model at each step, thereby opening the possibilities to improve the performance of traditional multi-class classification approaches. In this work, a novel hierarchical framework is proposed for early-stage detection of pancreatic cancer. Firstly, a hard hierarchical decision structure (HDS) coupled with feature engineering at each step provides a better performance as compared to traditional multi-class classification approaches. Secondly, a soft hierarchical decision structure (SDS) additionally provides confidence associated with predicted labels in the form of probability values for each class.

The major purpose of using computational methods for early pancreatic cancer detection is detecting the onset of pancreatic cancer in the group of chronic pancreatitis patients, which would permit a maximal time for successful treatment with emerging methods, such as immunotherapy [178]. The key contributions of this work are as follows:

- This work, for the first time, proposes the use of ultrasensitive nanobiosensors for protease/arginase detection and integrates it with an information fusion based hierarchical decision structure to detect pancreatic cancer (PC) at the localized stage by means of a simple Liquid Biopsy.

- HDS, coupled with appropriate feature engineering steps is proposed to improve the performance of traditional multi-class classification approaches. Results illustrate up to 17% improvement in performance with the proposed HDS scheme relative to conventional multi-class classification approaches.

- To better assist the clinician's decision-making and provide insights into the decision criteria driving the statistical methods, an SDS is developed to provide confidence scores associated with predicted labels, in the form of class probability values.

The decision labels and values of class probabilities obtained from HDS and SDS respectively support clinicians in recognizing situation where predictions of the computational models are uncertain. It helps them determine whether more tests are necessary for a more accurate diagnosis. In this manner, the proposed framework possesses the potential to serve as an effective decision support tool for early-stage PC detection.

8.1.1 Description of the dataset

The dataset resulting from protease/arginase activity quantified using ultrasensitive nanobiosensors consists of a set of eight biomarkers. Identified biomarkers were obtained from the NCBI Gene Expression Omnibus (GEO) database, which is public accessible. Biomarkers were proteases with significant differences in expression levels between two samples, a primary tumor sample and a healthy tissue sample, which both had to be in *Homo sapiens*. The features for each sample comprise of values corresponding to a panel of seven proteases and arginase, selected based on gene expression analysis using data from Unigene ID, Gene Symbol, NCBI GEO and Entrez Gene ID [162]. Protease/arginase activity for each identified biomarker was quantified in human serum samples obtained from the Biospecimen Repository Facility in the Cancer Center of the University of Kansas Medical Center [179]. The group size was as follows: "Healthy" volunteers ($n = 48$) and pancreatic cancer patients ($n = 31$), which was further divided into "Localized" (earlier stage) and "Metastatic" (later stage) pancreatic cancer. Localized pancreatic cancer samples signify the absence of any indication that the cancer has spread outside the pancreas, while metastatic pancreatic cancer indicates that it has spread to other parts of the body as well. Quantified protease/arginase activity was quantified in serum samples after 60 min incubation, and this dataset was then utilized to develop a computer prediction model.

Although the sample size for this research study is limited, it is important to note that we are proposing a unique, one-of-a-kind approach for early cancer detection. We believe that providing these initial results will stimulate more follow-on efforts in this direction creating a significant clinical impact. Specifically, for this study, the team was not able to

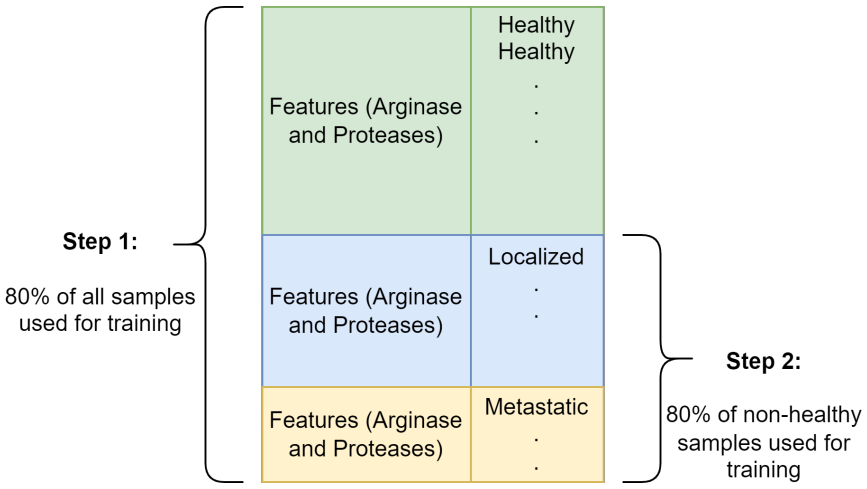


Figure 8.2: Formation of training set for early-stage detection of pancreatic cancer.

obtain more disease samples and had to work with the maximum number of samples the biospecimen repository at the University of Kansas Medical Center was able to recruit. It is imperative that the activity measured with this nanobiosensor technology depends strongly on the protocol and quality of the serum samples collected, which has been previously noticed by this team. For this reason, the better approach was to work with a smaller, but well-defined sample size instead of obtaining a larger sample size from different repositories to avoid multiple variables introduced during the comparison analysis. Furthermore, confidence intervals are provided for all the inferences in order to accommodate the sample size effects and better illustrate the power of the results.

The training set formation process for individual binary classifiers at respective hierarchical steps is illustrated in Figure 8.2. Eighty percent of all the instances in the dataset are randomly selected to train the binary classifier in the first hierarchical step. This step results in the isolation of samples belonging to "healthy" group. So, 80% of the remaining instances, i.e., the samples from "localized" and "metastatic" groups are selected at random to form the training set for binary classifier in the second hierarchical step. This strategy for preparing the training sets for individual binary classifiers is adopted due to limited number of samples available in the dataset.

8.1.2 Methodology

In this work, the problem of early-stage detection of pancreatic cancer is modelled as a multi-class classification problem. The data derived from the experiments consists of three classes, namely, "Healthy", "Localized" pancreatic cancer and "Metastatic" pancreatic cancer. Two information fusion based decision structures are proposed:

1. A HDS with specific feature engineering at each step for better performance relative to conventional classification approaches.

2. A SDS that provides confidence associated with predicted labels in the form of probability values for each class.

Hard Hierarchical Decision Structure

The fundamental premise of the proposed information fusion based HDS involves tailoring the statistically most significant features with appropriate weights to execute an efficient binary classification task at each hierarchical step. The proposed HDS is shown in Figure 8.3. The individual elements involved in building the HDS are described next.

Computing weights for features

The first step in the proposed HDS framework is to identify whether the given sample belongs to healthy (null hypothesis) or non-healthy (alternate hypothesis) group. The feature engineering in building the corresponding binary classifier involves appropriate weighing of all the features based on their relative importance. These weights are obtained based on the p-values of two-sample t-tests for all the features across the set of measurements obtained from "healthy" and "non-healthy" groups. The test decision values and p-values for the null hypothesis that measures if "healthy" and "non-healthy" groups belong to independent random samples from normal distributions with equal means, were evaluated for all the features. It was observed that the p-values are distributed over a wide range and possess a highly skewed distribution. Therefore, the associated probability operations can generate extremely small values that are difficult to represent with sufficient precision. This results in

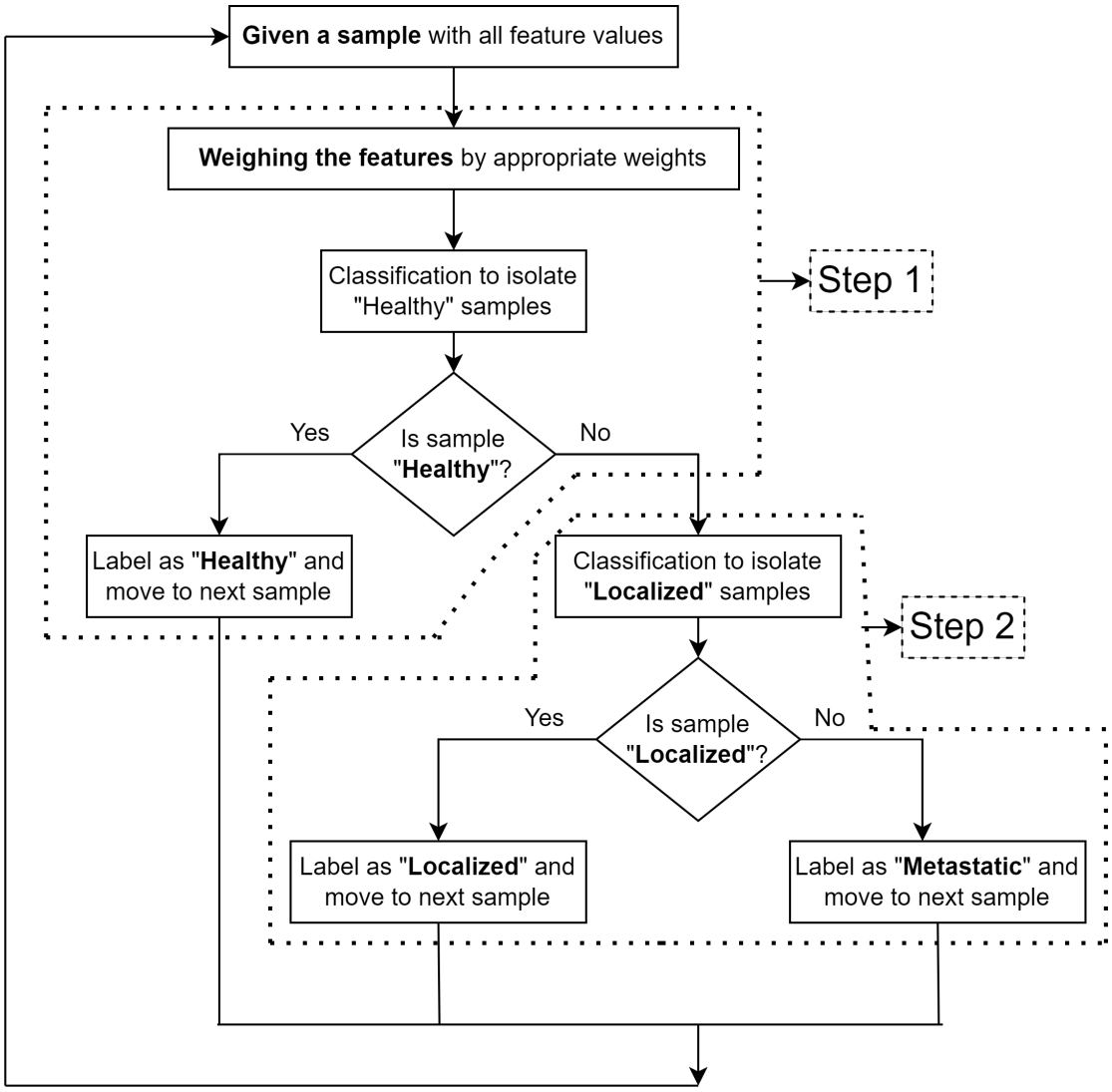


Figure 8.3: Proposed information fusion-based HDS framework.

Table 8.1: Computing weights for the features (A test decision value of 0 indicates a failure to reject the null hypothesis at 95% confidence level, a value of 1 indicates rejection of the null hypothesis at 95% confidence level; a rank of 1 indicates that the corresponding feature is most important and that of 8 indicates that the corresponding feature is least important).

Sr. No.	**Feature**	**Test Decision**	***p*-value**	**Rank**	$-\log_e p$	**Weight**
1	Arginase	1	3.41e-5	6	10.29	0.0950
2	Cat B	1	8.92e-10	2	20.84	0.1925
3	Cat E	1	9.61e-10	3	20.76	0.1918
4	MMP 1	0	0.0538	8	2.92	0.0270
5	MMP 3	1	0.0214	7	3.84	0.0355
6	MMP 9	1	5.79e-10	1	21.27	0.1965
7	Neutrophil Elastase	1	1.62e-6	5	13.34	0.1232
8	UpA	1	3.02e-7	4	15.01	0.1387

numerical errors like underflow or overflow. In order to avoid precision issues, the p-values are transformed to a logarithmic scale for better interpretation and analysis [180–182]. The negative values of natural logarithm of p-values, $-\log_e p$ is computed for all the features and scaled, as shown in equation (8.1) to obtain the corresponding feature weights.

$$w_i = \frac{-\log_e f_i}{\sum_{n=1}^{N} -\log_e f_n} \tag{8.1}$$

Here, w_i is the weight corresponding to feature f_i, $-\log_e f_i$ represents the negative value of natural logarithm of p-value corresponding to feature f_i and N is the total number of features. The p-values and computation of corresponding weights for all the features in the dataset under consideration are presented in Table 8.1.

Selecting features for each hierarchical step

If a given sample is identified as "non-healthy" in the first hierarchical step, the next step is aimed at determining the degree or extent of abnormality involved. The second hierarchical step determines if the given sample belongs to a "localized" or "metastatic" group. The corresponding binary classifier uses a subset of the features rather than using all the features obtained from experiments, as in the first hierarchical step. This feature engineering step identifies the most relevant features, thereby simplifying the models and making them easier to interpret. Moreover, this allows to have shorter training times and reduces overfitting.

The relevant features are identified by conducting a series of two-sample t-tests (as in the first hierarchical step) for localized vs. metastatic groups. The features exhibiting lowest p-values in hypothesis tests are selected as admissible features for corresponding binary classifier. For the dataset under consideration, Cat B, Cat E, MMP 3 and UpA are selected as admissible features for binary classifier in the second hierarchical step.

Soft Hierarchical Decision Structure

While the HDS offers a three-class classifier, it does not provide any information regarding the confidence associated with the decisions. This drawback is addressed in the proposed SDS framework that provides confidences associated with the predicted labels in the form of probability values for each class. The proposed SDS is shown in Figure 8.4. It is basically an extension of the HDS, where the prediction for each sample is accompanied with the probability values of that sample being affiliated to each of the three classes. The differences between these probability values provide an indication of confidence associated with the corresponding prediction. For a given instance, if the probability value corresponding to one of the classes in significantly higher than the rest, the confidence associated with such a prediction would be HIGH. On the other hand, if there is no significant difference between the probability values corresponding to all the classes, the associated confidence would be LOW. This framework helps the doctors determine whether additional tests are required for proper diagnosis.

All steps in the SDS are probabilistic extensions of the HDS. For example, the first step in SDS results in two values indicating probabilities of the given sample being "healthy" or "non-healthy", represented by $P(H)$ and $P(\overline{H})$ respectively. These values are essentially the probability estimates of classification model trained at first hierarchical step for a given sample and are obtained using predict_proba() method of trained scikit-learn [183] models. The second hierarchical step evaluates the probabilities of the given sample being "localized" or "non-localized", given the condition that it belongs to "non-healthy" group, represented by $P(L|\overline{H})$ and $P(\overline{L}|\overline{H})$ respectively. These values are probability estimates of classification

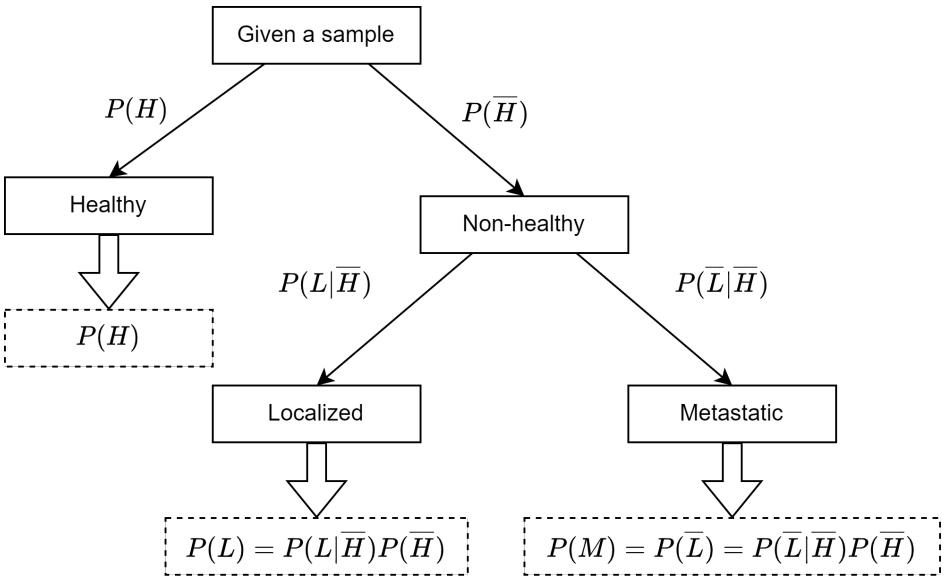


Figure 8.4: Proposed information fusion-based SDS framework.

model trained at second hierarchical step for a given sample. As a result, the probabilities of a sample being "localized" or "metastatic" is evaluated based on equations (8.2) and (8.3) respectively.

$$P(L) = P(L|\overline{H})P(\overline{H}) \tag{8.2}$$

$$P(M) = P(\overline{L}) = P(\overline{L}|\overline{H})P(\overline{H}) \tag{8.3}$$

8.1.3 Results and Discussion

Hard Hierarchical Decision Structure

The proposed framework is evaluated by training a series of hierarchical classification models by considering several combinations of binary classifiers in all the three hierarchical steps, indicated in Figure 8.3. The classification methods considered for individual binary classifiers include: (i) Gaussian Naïve Bayes (GNB) [184], (ii) Decision Tree (DT) [185], (iii) SVM [186], (iv) kNN [187], (v) RF Classifier [185, 187] and Logistic Regression (LR) [188]. In order to avoid overfitting, k-fold ($k = 5$) cross validation technique is used as a resampling method

Table 8.2: Combinations of classification methods exhibiting an overall mean accuracy score of more than 85%.

Sr. No.	Classification Method		Accuracy Score	Sensitivity	Specificity
	Step 1	Step 2			
1	GNB	DT	(87.34 ± 4.70)%	0.76 ± 0.10	0.84 ± 0.04
2	kNN	DT	(89.87 ± 6.28)%	0.81 ± 0.06	0.89 ± 0.06
3	kNN	kNN	(91.14 ± 3.94)%	0.85 ± 0.08	0.93 ± 0.04
4	kNN	SVM	(92.40 ± 3.26)%	0.93 ± 0.08	0.95 ± 0.06
5	RFC	DT	(88.61 ± 4.66)%	0.88 ± 0.04	0.91 ± 0.04
6	RFC	GNB	(86.07 ± 6.94)%	0.78 ± 0.06	0.88 ± 0.02
7	RFC	kNN	(89.87 ± 5.04)%	0.89 ± 0.02	0.86 ± 0.08
8	SVM	SVM	(88.61 ± 3.36)%	0.87 ± 0.08	0.90 ± 0.06
9	SVM	GNB	(87.48 ± 5.52)%	0.80 ± 0.06	0.88 ± 0.02
10	DT	GNB	(86.07 ± 7.76)%	0.78 ± 0.04	0.82 ± 0.08

Table 8.3: Confusion Matrix – HDS (Step 1: kNN; Step 2: SVM).

		Predicted Class		
		Healthy	**Localized**	**Metastatic**
True Class	**Healthy**	**46**	1	1
	Localized	1	**18**	0
	Metastatic	1	2	**9**

for training and evaluating the performance of classification models. The combinations of classification methods exhibiting an overall mean accuracy score of more than 85% are reported in Table 8.2. The sensitivity and specificity of all model combinations are also indicated. The 95% confidence intervals for evaluation metrics (accuracy score, sensitivity and specificity) are represented using mean and standard deviation of k-fold cross-validated values.

The training sets were formed and evaluation was performed over all the instances in the dataset under consideration. In Table 8.2, it can be observed that the best performance (overall mean accuracy score of 92.40%) is obtained using kNN for binary classification in first hierarchical step and SVM in the second step. Moreover, the sensitivity and specificity scores for this case are observed to the most favorable as compared to all other combinations of classification methods. The corresponding confusion matrix is presented in Table 8.3.

On the contrary, the maximum mean classification accuracy obtained from conventional multi-class classification approach using individual classification methods is 74.66%, as indicated in Table 8.4. The 95% confidence intervals associated with the predictions of statistical

Table 8.4: Performance obtained using conventional multi-class classification approaches.

Sr. No.	Classification Method	Accuracy Score
1	GNB	(70.48 ± 7.50)%
2	DT	(68.52 ± 11.22)%
3	SVM	(71.21 ± 5.74)%
4	kNN	(74.66 ± 4.28)%
5	RFC	(69.18 ± 7.82)%
6	LR	(72.95 ± 9.16)%

models are reported using mean and standard deviation of k-fold ($k = 5$) cross-validated accuracy scores. This demonstrates that the HDS framework outperforms the conventional multi-class classification approaches for early-stage detection of pancreatic cancer. The superior performance of proposed HDS framework is primarily attributed to the following reasons: (i) the features are weighed in the first hierarchical step based on their distinguishing ability, unlike traditional multi-class classification approaches which give equal importance to all the features; (ii) only a subset of features which are able to confidently differentiate between localized and metastatic PC are considered in the second hierarchical step, instead of accounting for all the features irrespective of their differentiating capability; (iii) splitting a multi-class classification problem into step-wise binary classification tasks allows for a more simplified feature representation and better learning.

Soft Hierarchical Decision Structure

The proposed SDS framework supports computation of confidences associated with the predicted labels in the form of probability values for each class. An example for a correct and incorrect prediction are shown in Figure 8.5 and Figure 8.6 respectively.

The instance shown in Figure 8.5 is correctly classified as "Healthy". It can be seen that the probability of this sample belonging to "Healthy" class is significantly higher than those of the other classes. In such a situation, the clinician can have sufficiently high confidence on the model prediction and it can be concluded that no further tests are required. In contrast, the instance shown in Figure 8.6 actually belongs to a "Healthy" class but is misclassified as "Metastatic". Additionally, it can be observed that the differences in probabilities of "Healthy" and "Metastatic" classes is not as significant as in the instance demonstrated in

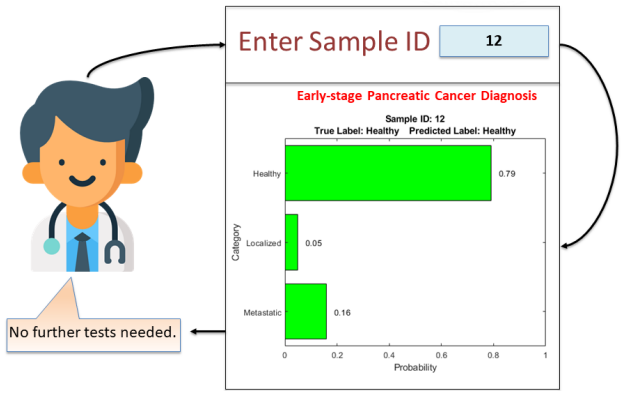


Figure 8.5: Example of correct prediction - SDS.

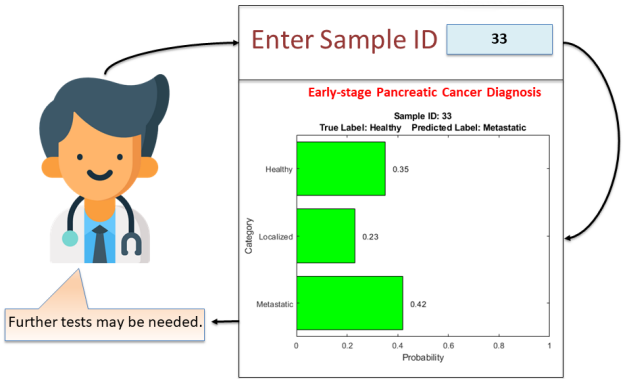


Figure 8.6: Example of incorrect prediction - SDS.

Figure 8.5. One of the fundamental limitations of standard AI-based decision-making models is that they attempt to impose a strict conclusion in the form of an output by selecting the most appropriate option among all the possibilities. In such a scenario, the confidence and faithfulness towards predictions of these computational models is disputable, particularly for crucial applications such as medical diagnosis. The proposed SDS framework overcomes this shortcoming by specifying confidence associated with model predictions in the form of class probability values. This information helps the clinician perceive the lack of confidence in the model predictions and nudge them to possibly prescribe further tests prior to diagnosis. In a sense, the SDS builds off the HDS and makes it more "explainable" to the end user.

8.2 Case Study 2: Early-stage Detection of Ovarian Cancer

Ovarian Cancer (OC) stands as the most fatal among all gynecologic malignancies. It ranks as the seventh most frequently diagnosed cancer and the eighth leading cause of cancer-related deaths among women worldwide. Estimates provided by the American Cancer Society suggest that in 2023, there will be approximately 19,710 new cases of ovarian cancer and 13,270 associated deaths [189–191]. OC poses a significant threat with no efficient screening methods available. Around 70% of OC patients receive diagnoses at advanced stages, leading to a grim prognosis, despite aggressive and prompt therapeutic interventions [192]. OC refers to a diverse collection of carcinomas originating in ovarian tissue. These carcinomas exhibit distinct clinicopathological and molecular characteristics, as well as unique tissue origins [193]. Ovarian High Grade Serous Carcinoma (HGSC) represents the predominant subtype, comprising approximately 70–80% of all diagnosed ovarian cancers worldwide. The heightened mortality associated with HGSC is largely attributed to the challenge of early detection, a crucial factor for effective clinical interventions. Symptoms of HGSC typically manifest insidiously, often lacking noticeable clinical indications until advanced stages. Common symptoms include bloating, pelvic or abdominal discomfort, and menstrual irreg-

ularities. Efforts have been made by researchers to devise diagnostic models for ovarian cancer symptoms, aiming to enhance OC detection. Despite these endeavors, symptoms are reported to occur across all stages of the disease, yet they tend to be non-specific [194]. Consequently, approximately 70% of high-grade serous carcinoma (HGSC) cases evade diagnosis until reaching stages III-IV [195]. As a result, the long-term survival prospects for women diagnosed with ovarian cancer remain bleak, with a mere 5-year survival rate of 49% [196].

Routine screening for early detection has proven highly beneficial for various types of cancer. For instance, the implementation of the Papanicolaou (Pap) test has led to a significant decrease in both incidence and mortality rates of cervical cancer by over 75% in screened populations in the United States. Likewise, colonoscopy screening has been associated with a notable 70% reduction in mortality risk for colorectal cancer [197]. Regrettably, effective screening for ovarian cancer remains elusive as there is presently no established strategy for early detection [198]. Presently, two approaches exist to screen for ovarian cancer before symptoms manifest or become detectable during routine gynecologic examinations. One method involves a blood test to detect elevated levels of a protein known as Cancer Antigen 125 (CA-125) [199]. The alternative method involves conducting an ultrasound examination of the ovaries. However, studies have yet to demonstrate that either technique significantly improves survival rates when employed in women at average risk. Consequently, routine screening for ovarian cancer is presently not recommended by the United States Preventive Services Task Force (USPSTF) due to the assessment that "the potential harms outweigh the potential benefits" [200, 201]. Therefore, there exists a pressing need for precise screening and diagnostic methods for ovarian cancer. Consequently, clinicians and researchers have been striving to innovate and develop novel approaches to identify early-stage disease utilizing a range of diverse biomarker sources. The objective of this study is to devise a viable, non-invasive, and economical detection methodology with the capability to identify ovarian cancer at its earliest stage via a simple blood test, commonly referred to as a liquid biopsy. By following a similar procedure as described in Section 8.1.1, the Bossmann group (our collaborators at The University of Kansas Medical Center) established a panel of seven proteases including MMPs, CTSs and ADAMs by means of graphene-based nanobiosensors

nanobiosensors to detect activity of OC proteases in serum. The data is based on experiments conducted with 146 participants, classified into 3 classes (Healthy, Localized and Metastatic) on the basis of an enzymatic signature consisting of MMP3, MMP24, MMP28, CTSK, ADAM10-12, ADAM15 and ADAM17.

8.2.1 Methodology

In this work, the problem of early-stage detection of ovarian cancer is modelled as a multi-class classification problem. The data derived from the experiments consists of three classes, namely, "Healthy", "Localized" ovarian cancer and "Metastatic" ovarian cancer. This work proposes an active explainable decision framework for early-stage detection of OC. AL is implemented with the following variants for base decision model: (i) Hierarchical decision model with kNN as base classifier (similar to the one discussed in Section 8.1.2), (ii) 2-layer neural network with linear activations. US query strategy is employed to select instances for annotation during the iterative querying process in AL environment. It is worthwhile to note that the use of aforementioned base decision models is just an illustration and the proposed framework is applicable across any base classification framework. In this strategy, the instances which minimize the classifier uncertainty are selected for annotation during each query step, as indicated in eq. (6.16). 20% of the total instances in the dataset are selected randomly to create an initial labeled dataset, which is used to train an initial ML-based classification model. Further, 30% of the total instances are used for querying iteratively (one query per iteration), and the classification model is updated after each query step.

In order to explain the decisions predicted by the proposed framework, feature relevance explanation is used as post-hoc explainability technique for the decision models. This process elucidates the internal operations of a model by calculating a relevance score for each of the associated features in the dataset. These scores measure the impact (sensitivity) a feature exerts on the model's output. Comparing scores across various variables reveals the significance attributed by the model to each variable in generating its output. The feature relevance methods serve as an indirect approach to elucidate the predictions of a decision

Table 8.5: Confusion Matrix with hierarchical framework as base decision model.

		Predicted Class		
		Healthy	**Localized**	**Metastatic**
True Class	**Healthy**	**47**	1	2
	Localized	3	**42**	1
	Metastatic	0	2	**48**

model [202]. Specifically, SHapley Additive exPlanations (SHAP) methodology [203] is employed to address the issue of explainability. SHAP is a unified framework that assigns a significance value to each feature for a specific prediction task using a novel category of additive feature importance metrics. An additional advantage is that SHAP values remain constant unless there is a modification in the contribution of a feature. This suggests that SHAP values are model-agnostic and offer a reliable interpretation of the predictions of a decision model, regardless of alterations in its architecture or parameters. Moreover, it is robust to missing data and mitigates the risk of irrelevant features distorting the interpretation of the decision model.

8.2.2 Results and Discussion

The proposed AL methodology was evaluated by considering both the variants for base decision model: (i) Hierarchical decision framework with kNN as base classifier, and (ii) 2-layer neural network with linear activations. The corresponding confusion matrices are presented in Table 8.5 and Table 8.6, respectively. It can be observed that a classification accuracy of 93.83% is obtained when hierarchical decision framework with kNN is employed as thee base decison model within AL environment. On the other hand, a 2-layer neural network with linear activations yields an accuracy of 94.52%. The specificity is same with both the variants of decision models. Moreover, the sensitivity is slightly better with 2-layer neural network (0.94), as compared to tat obtained using hierarchical decision framework with kNN (0.93) as the base decision model. This suggests that a 2-layer neural network with linear activations is able to learn better representation in the proposed AL framework.

The feature relevance scores were computed for all the samples in the dataset, based on

Table 8.6: Confusion Matrix with 2-layer neural network as base decision model.

		Predicted Class		
		Healthy	**Localized**	**Metastatic**
True Class	**Healthy**	**47**	2	1
	Localized	0	**44**	2
	Metastatic	1	2	**47**

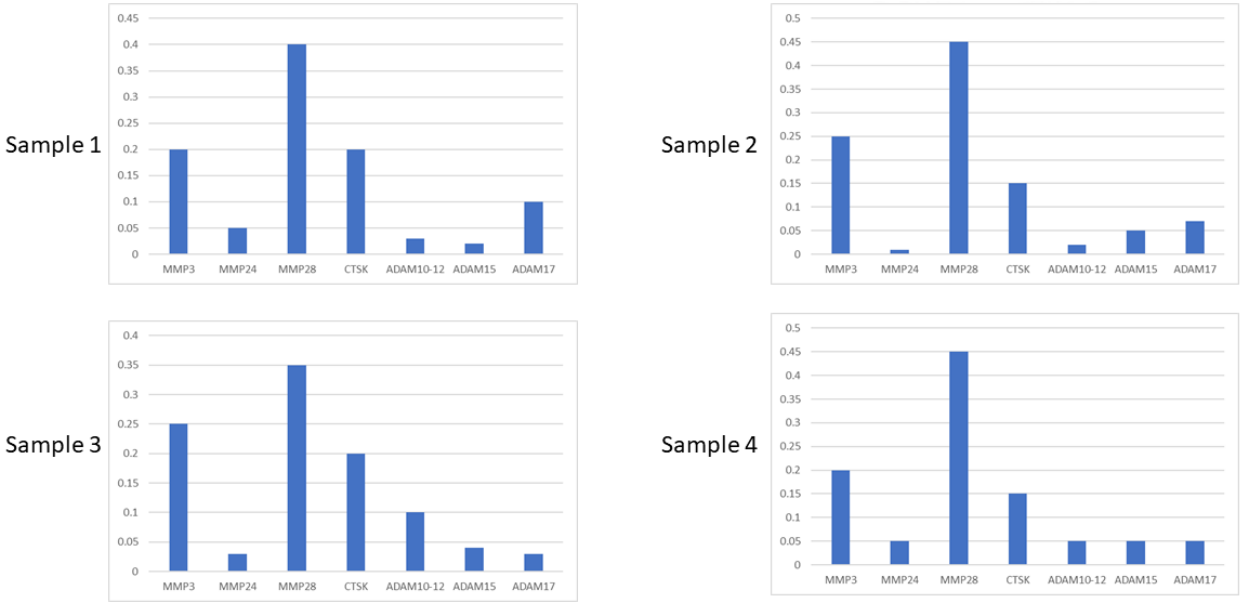


Figure 8.7: Feature relevance scores for samples in "Healthy" class.

the SHAP methodology elaborated in Section 8.2.1. A few samples from each of the classes, i.e., "Healthy", "Localized" ovarian cancer and "Metastatic" ovarian cancer are presented in Figure 8.7, Figure 8.8, and Figure 8.9, respectively. It can be observed that the top three feature relevance scores are consistent within each class, for example, MMP3, MMP28, CTSK exhibit the highest feature relevance scores for samples in "Healthy" class. Similarly, MMP3, MMP24, ADAM10-12 possess the highest feature relevance scores for "Localized" class, and MMP28, ADAM15, ADAM17 for the "Metastatic" class. The consistency in these feature relevance scores suggest the high importance of these features over others in order to explain the predictions of the decision models. This information helps the clinician to judge the importance of different proteases during medical diagnosis and nudge the possibility of prescribing additional tests before making a final decision.

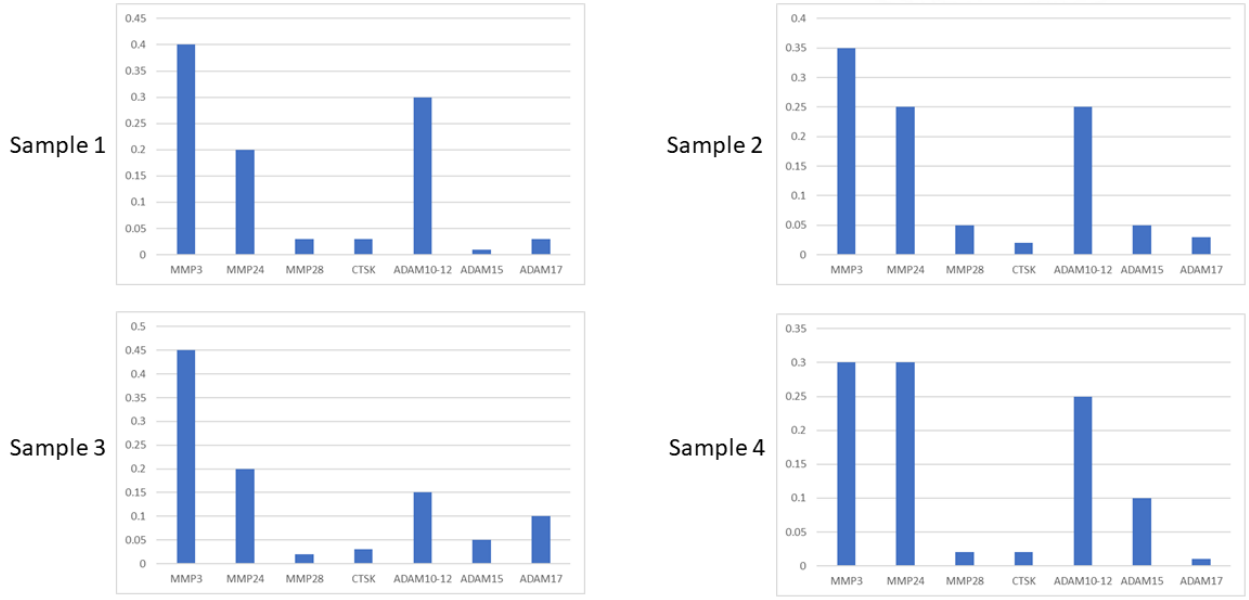


Figure 8.8: Feature relevance scores for samples in "Localized" class.

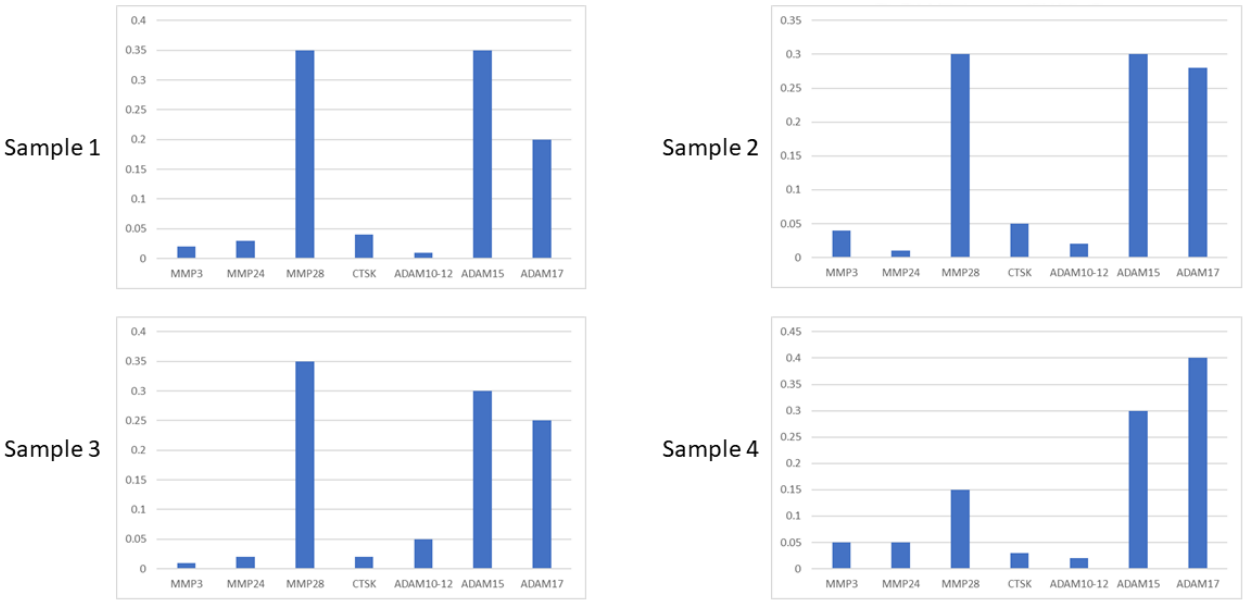


Figure 8.9: Feature relevance scores for samples in "Metastatic" class.

8.3 Summary

In this chapter, two case studies are presented to demonstrate different ways of explaining the predictions made by decision models. The first case study focuses on early-stage detection of PC. The use of ultrasensitive nanobiosensors for protease/arginase detection is combined with information fusion based statistical framework to detect PC at the localized stage by means of a simple Liquid Biopsy. The proposed hierarchical SDS framework provides confidences associated with the predicted labels in the form of probability values for each class, which makes the decisions more "interpretable". This information can be used to clinicians in order to perceive the lack of confidence in model predictions and to examine if any further tests are required before making a final decision. The prime advantage of using such computational methods for detection of pancreatic cancer during early-stage is detecting the onset of pancreatic cancer in the group of chronic pancreatitis patients, which would allow a maximal time for successful treatment with emerging methods like immunotherapy. The second case study on early-stage detection of OC incorporates feature relevance technique to explain the predictions of corresponding decision models. This provides a feature relevance score for all the features corresponding to a sample under consideration. This information aids clinicians in assessing the significance of various proteases during medical diagnosis, thereby enabling them to consider the necessity of additional tests before reaching a definitive decision.

The next chapter introduces a novel framework to leverage the knowledge of input and output uncertainties in decision models, modify training objectives and improve the robustness and predictive performance of decision models. The knowledge of uncertainty is also used to derive upper and lower bounds for label complexity, in the context of AL.

Chapter 9

Uncertainty-guided Active Learning

NN-based decision models have enabled a plethora of AI applications in different domains of science and engineering over the last decade. Regardless of their stellar predictive performance, there are two key challenges that limit their adaptability and reliability: (i) requirement of massive amounts of labeled training data, and (ii) inability to quantify uncertainties in the model predictions [204]. Semi-supervised learning approaches like Active Learning (AL) have proven to be beneficial in training decision models using much fewer labelled instances when compared to conventional supervised learning methods [205]. This is accomplished by strategic selection of the most informative samples from the unlabeled pool of data by optimizing specific AL heuristics. It allows learning an efficient decision model within the available annotation budget, either in terms of cost or time [53, 81, 206]. Owing to the widespread applicability of these computational models, it is extremely crucial to quantify the confidence associated with their predictions. The issue of improper calibration (i.e., over-confidence or under-confidence) of the decision models becomes more important in safety-critical applications like healthcare.

In order to overcome the aforementioned limitation, uncertainty quantification in NN-based decision models has gathered significant research attention lately. The fundamental goal of such approaches is to provide probability distributions of model predictions or the confidence intervals associated with them, rather than offering just the point estimates. The

uncertainties can be broadly divided into two categories: (i) aleatoric uncertainty, and (ii) epistemic uncertainty. Aleatoric uncertainty is defined as the intrinsic randomness of data caused by noisy or faulty measurements. It can be captured via uncertainty associated with the feature vectors of individual data samples. Epistemic uncertainty refers to the scientific uncertainty in a model caused by its inability to completely comprehend the underlying process. It can be captured via uncertainty associated with the parameters and activation functions of neurons related to the NN model. A variety of methodologies based on the use of Bayesian Neural Networks (BNN), ensemble models and deep Gaussian processes have been used to quantify epistemic uncertainty [207]. On the other hand, aleatoric uncertainty has been quantified using approached based on deep discriminative models and deep generative models [207]. Once the uncertainty in NN is quantified, this knowledge can be leveraged to modify training objectives, thereby improving the robustness and predictive performance of decision models. This work proposes, for the first time, a novel framework to leverage the knowledge of prediction/classification uncertainty to guide querying process in the context of AL.

9.1 Related Work

The predictions of NN-based decision models are impacted by numerous factors that lead to uncertainties in both data and model. The presence of multiple sources of uncertainty makes it challenging to completely eliminate uncertainties in neural networks, thereby making it impractical for a majority of safety critical applications. In practical scenarios that include the use of real-world data, it is common for the training data to represent only a portion of the entire range of possible input data. Consequently, it is sometimes inevitable to encounter a discrepancy between the domains of NN and the unknown actual data [204]. Nevertheless, it is challenging to precisely assess the uncertainty of NN predictions due to the inherent difficulty in modelling the possibly unknown nature of various uncertainties involved. Hence, the quantification and exploitation of uncertainties in NN-based decision models has been studied extensively in the literature recently.

Uncertainty quantification methods can be broadly classified into four categories [204]: (i) single deterministic methods, (ii) Bayesian methods, (iii) Ensemble methods and (iv) Test-time augmentation methods. In single deterministic methods, the predictive uncertainty is derived from a single forward pass within a deterministic network. These methods can be internal or external. The internal deterministic methods involve predicting the distribution parameters over the predictions rather than just the point maximum-a-posteriori estimates [208–211]. On the other hand, the external deterministic methods are model-agnostic and can be integrated with already trained networks [212–214]. Although single deterministic methods are computationally efficient in terms of implementation, they depend on a single opinion, thereby making them extremely sensitive towards training data, methodology as well as model architecture. Bayesian methods infer the probability distribution over the network parameters by employing Bayesian learning instead of maximum likelihood principles. In these methods, the posterior distribution is often intractable for a majority of the prior posterior pairs. Consequently, approximation techniques like variational inference [215, 216], sampling approaches [217, 218] and Laplace approximation [219, 220] are employed to approximate the Bayesian inference.

Ensemble methods generate predictions by aggregating the decisions obtained from several ensemble members. The synergistic effects across several decision models are utilized to achieve a more comprehensive generalization. This is based on the argument that a collection of decision models is more likely to provide superior conclusions compared to a single decision model [221]. In the recent years, different variants of ensemble approaches like Deep Ensemble [222], Hyperparameter Ensemble [223], Batch Ensemble [224] and Multi-Input Multi-Output [225] have been proposed for uncertainty quantification in NN-based decision models. Since each ensemble member is independent, there is a linear growth in memory and computational resources when more members are added. This applies to both the training and testing phases. Thus, implementing ensemble methods is constrained in several practical applications due to limitations in computation power, memory capacity, time sensitivity, and the inclusion of huge networks with high inference time. The test-time augmentation methods are motivated by a combination of adversarial examples and ensemble methods [226].

The fundamental approach involves generating several test samples from each original test sample through the application of data augmentation techniques. Subsequently, all of these augmented samples are tested to calculate a prediction distribution, which serves as a means to quantify uncertainty. The underlying concept of this approach is that by incorporating augmented test samples, it becomes possible to examine several perspectives, hence enabling uncertainty quantification [227–229].

In the context of AL, similar approaches have been recently adopted for uncertainty quantification [230–233]. However, none of the works in the literature address the problem of leveraging input and output uncertainties in NN-based decision models to either guide the querying process in AL or modify the training objectives. In this paper, for the first time, the knowledge of predictive uncertainty is leveraged to guide the process of informative sample selection in AL. The lower bounds and upper bounds for label complexity are also derived.

In this work, a novel framework is proposed to leverage the knowledge of uncertainty to guide querying process in AL. Given a neural network-based decision model and a tolerable threshold on the uncertainty of its predictions, the prediction uncertainty is first propagated back to the input side using the Assumed Density Filtering (ADF) approach. This knowledge is then leveraged to guide the informative sample selection via querying in AL. The lower and upper bounds for label complexity are also derived in the context of AL approach considered in this work. The proposed methodology is generic and can be applied to any AL setup, irrespective of the query strategy and base decision model. Such an approach of utilizing uncertainty information to support AL has not been reported in the literature related to uncertainty in NN so far and is a novel contribution of this work. The efficacy of the proposed framework has been experimentally demonstrated across both classification and regression problem setups.

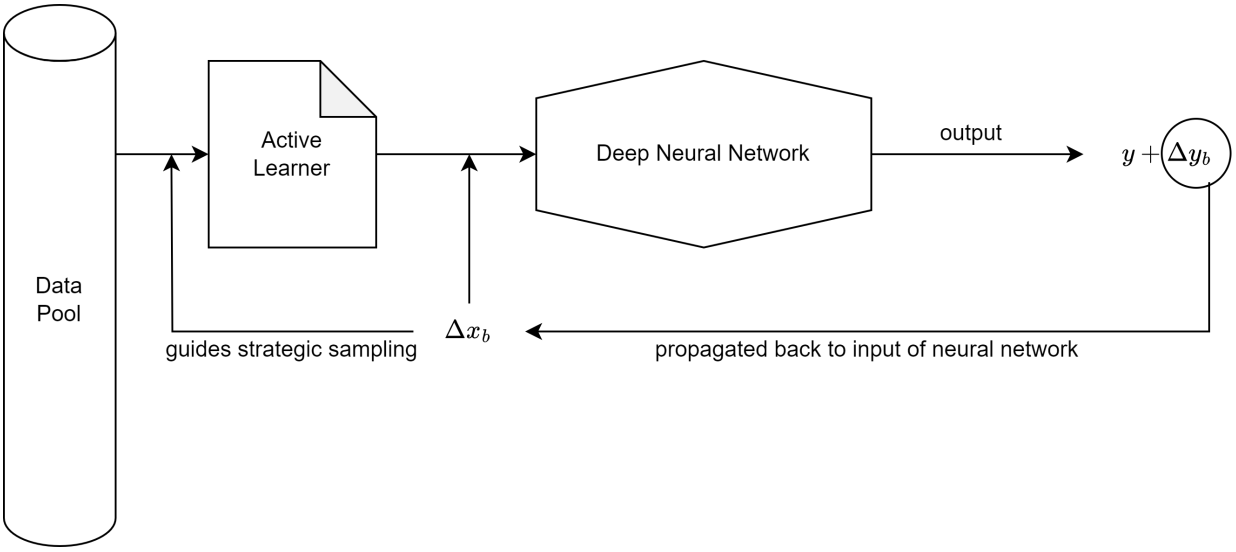


Figure 9.1: Proposed framework for uncertainty-guided Active Learning.

9.2 Methodology and Main Results

This work presents a novel framework to use the knowledge of uncertainty in model predictions to guide the AL sampling process, which in turn helps in fine-tuning the decision model. Given a neural network-based decision model and a tolerable threshold on the uncertainty in its predictions, the proposed framework involves two major components: (i) propagating the tolerable threshold on uncertainty of the model predictions to the input side; and (ii) using the input uncertainty to guide the informative sample selection via querying in AL. This is illustrated in Figure 9.1. Here, $\Delta \boldsymbol{y}_b$ represents output uncertainty (i.e., uncertainty associated with the predictions of NN-based decision models) and $\Delta \boldsymbol{x}_b$ represents input uncertainty (i.e., uncertainty associated with the input features fed to the neural networks). In practice, the sources of these uncertainties are: (a) intrinsic randomness of the data due to noisy or inaccurate measurements, (b) incompetency of the models to completely explain the underlying process. In this work, we put a tolerable threshold on uncertainty of the model predictions ($\Delta \boldsymbol{y}_b$), and then systematically propagate it to across the neural network layers to obtain $\Delta \boldsymbol{x}_b$.

Each of the components in proposed methodology is discussed next.

9.2.1 Propagation of uncertainty in Deep Neural Networks

A Deep Neural Network (DNN) can be represented as a cascade of non-linear layers, as shown in (9.1). That is,

$$\begin{aligned} \boldsymbol{y} &= \boldsymbol{f}(\boldsymbol{x};\boldsymbol{\theta}) \\ &= \boldsymbol{f}^{(l)}\left(\boldsymbol{f}^{(l-1)}\left(...\boldsymbol{f}^{(1)}\left(\boldsymbol{x};\boldsymbol{\theta}^{(1)}\right)\right)\right) \end{aligned} \tag{9.1}$$

where, l is the number of layers in DNN, x represents the input features fed to DNN and y refers to the prediction of DNN.

Each layer i of a DNN is represented by $\boldsymbol{f}^{(i)}\left(\boldsymbol{z}^{(i-1)};\boldsymbol{\theta}^{(i)}\right)$, i.e., a non-linear transformation of intermediate activations $\boldsymbol{z}^{(i-1)}$, parameterized by $\boldsymbol{\theta}^{(i)}$ with $\boldsymbol{z}^{(0)} = \boldsymbol{x}$. In the context of probabilistic feed-forward neural network, the point predictions y can be written in the form of probability distributions, as shown in (9.2). This helps to attribute a probability distribution to all the possible predictions $\boldsymbol{y}$.

$$\boldsymbol{f}(\boldsymbol{x};\boldsymbol{\theta}) \equiv p(\cdot|\boldsymbol{x}) \tag{9.2}$$

In this work, the predictive distribution $p(\boldsymbol{y}|\boldsymbol{x})$ is restricted to be parametric and the parameters are assumed to be central moments of a Gaussian distribution, where variance represents the uncertainty surrounding the mean. In case of non-Gaussian distributions, it needs to be approximated as Gaussian to retain a tractable form. Considering a standard architecture for propagating point activations [234], the joint probability density of activations in all the DNN layers can be written as:

$$p(\boldsymbol{z}^{(0:l)}) = p(\boldsymbol{z}^{(0)}) \prod_{i=1}^{l} p(\boldsymbol{z}^{(i)}|\boldsymbol{z}^{(i-1)}) \tag{9.3}$$

$$p(\boldsymbol{z}^{(i)}|\boldsymbol{z}^{(i-1)}) = \delta\left[\boldsymbol{z}^{(i)} - \boldsymbol{f}^{(i)}(\boldsymbol{z}^{(i-1)})\right] \tag{9.4}$$

where $\delta[\cdot]$ is the Dirac delta function. The uncertainty in the input features can be modeled as the true values perturbed by white Gaussian noise. Therefore,

$$p(\boldsymbol{z}^{(0)}) = \prod_j \mathcal{N}(z_j^{(0)}|x_j, \sigma_n^2) \tag{9.5}$$

ADF is employed to propagate this aleatoric uncertainty through the layers of the neural network. The ADF approach is chosen due to its low computational demands [234, 235]. This process of uncertainty propagation involves estimating a tractable approximation of neural network activations by consolidating one layer at a time, as shown in (9.6). Beginning with independent Gaussian activations at the input layer, i.e., $q(\boldsymbol{z}^{(0)}) = p(\boldsymbol{z}^{(0)})$, the subsequent layer activations can be approximated as,

$$p(\boldsymbol{z}^{(0:l)}) \approx q(\boldsymbol{z}^{(0:l)}) = q(\boldsymbol{z}^{(0)}) \prod_{i=1}^{l} q(\boldsymbol{z}^{(i)}) \tag{9.6}$$

$$p(\boldsymbol{z}^{(i)}) = \prod_j \mathcal{N}(z_j^{(i)}|\mu_j^{(i)}, v_j^{(i)}) \tag{9.7}$$

where, $\left(\mu_j^{(i)}, v_j^{(i)}\right)$ represents the value of activation and its variance corresponding to neural unit j at the layer i. Essentially, the activation distribution $q(\boldsymbol{z}^{(i-1)})$ is non-linearly transformed by subsequent layers $\boldsymbol{f}^{(i)}$ into an output with probability distribution $p(\boldsymbol{z}^{(i)}|\boldsymbol{z}^{(i-1)})q(\boldsymbol{z}^{(i-1)})$. Consequently, the joint density of all the activations upto layer i can be approximated as:

$$\tilde{p}(\boldsymbol{z}^{(0:i)}) = p(\boldsymbol{z}^{(i)}|\boldsymbol{z}^{(i-1)}) \prod_{j=0}^{i-1} q(\boldsymbol{z}^{(j)}) \tag{9.8}$$

Further, the best approximate probability distributions $\tilde{q}(\boldsymbol{z}^{(0:i)})$ are estimated by minimizing the KL divergence with $\tilde{p}(\boldsymbol{z}^{(0:i)})$. The next step is to perform moment matching between $\tilde{p}(\boldsymbol{z}^{(0:i)})$ and $\tilde{q}(\boldsymbol{z}^{(0:i)})$. Any layer $\boldsymbol{z}^{(i)} = \boldsymbol{f}^{(i)}\left(\boldsymbol{z}^{(i-1)}; \boldsymbol{\theta}^{(i)}\right)$ can be then represented as an uncertainty propagation layer by matching the central moments at first and second orders, as:

$$\boldsymbol{\mu}_z^{(i)} = \mathbb{E}_{q(\boldsymbol{z}^{(i-1)})}\left[\boldsymbol{f}^{(i)}\left(\boldsymbol{z}^{(i-1)}; \boldsymbol{\theta}^{(i)}\right)\right] \tag{9.9}$$

$$\boldsymbol{v}_z^{(i)} = \mathbb{V}_{q(\boldsymbol{z}^{(i-1)})}\left[\boldsymbol{f}^{(i)}\left(\boldsymbol{z}^{(i-1)}; \boldsymbol{\theta}^{(i)}\right)\right] \tag{9.10}$$

where $\mathbb{E}[\cdot] and \mathbb{V}[\cdot]$ denote expectation and variance, respectively.

In a DNN, the layers with linear activations are indicated as:

$$f(\boldsymbol{z}) = \boldsymbol{W}\boldsymbol{z} + \boldsymbol{b} \tag{9.11}$$

where

$$\boldsymbol{z}^{(i)} = \boldsymbol{W}\boldsymbol{z}^{(i-1)} + \boldsymbol{b} \tag{9.12}$$

This can be written in the form of uncertainty propagation layer as:

$$\boldsymbol{\mu}_z^{(i)} = \boldsymbol{W}\boldsymbol{z}^{(i-1)} + \boldsymbol{b} \tag{9.13}$$

$$\boldsymbol{v}_z^{(i)} = (\boldsymbol{W} \cdot \boldsymbol{W})\boldsymbol{v}_z^{(i-1)} \tag{9.14}$$

This leads to the following result: Given an uncertainty bound ($\Delta \boldsymbol{y}_b$) at the output of a DNN, the mean and variance of the activations at each preceding layer can be written as:

$$\boldsymbol{\mu}_z^{(i-1)} = \boldsymbol{W}^{-1}[\boldsymbol{\mu}_z^{(i)} - \boldsymbol{b}] \tag{9.15}$$

$$\boldsymbol{v}_z^{(i-1)} = (\boldsymbol{W} \cdot \boldsymbol{W})^{-1}\boldsymbol{v}_z^{(i)} \tag{9.16}$$

with $\boldsymbol{\mu}_z^{(l)} = \boldsymbol{y}$ and $\boldsymbol{v}_z^{(l)} = \Delta \boldsymbol{y}_b$

9.2.2 Leveraging uncertainty to guide Active Learning

After propagating the tolerable threshold for uncertainty in the model predictions to the input side, the next step is to leverage this knowledge to guide informative sample selection from the unlabeled pool of data. This is performed in two steps:

1. Sub-sampling the querying space: First, we calculate the Mahalanobis distance [236] between each point in the unlabeled pool of data and the initial labeled dataset (X_p) in the context of Active Learning. The Mahalanobis distance is computed as

$$d_M(x, X_p) = \left[(x - \overline{x_p})^T C^{-1} (x - \overline{x_p})\right] \tag{9.17}$$

where $\overline{x_p}$ represents the mean of X_p and C is the sample covariance matrix. This metric is a measure of the number of standard deviations a given data point is away from the distribution of X_p. Based on the value of $\Delta \boldsymbol{x}_b$, we put a threshold (d_T) on the Mahalanobis distance. All the points with distance less than d_T comprise the sub-sampled querying space for AL. Consequently, the disagreement region in eq. (2.20) can be modified to include this threshold, as shown below:

$$DIS(\mathcal{H}) = \{x \in X : \exists h, g \in \mathcal{H} \; s.t. \; h(x) \neq g(x) \cap [d_M(x, X_p) \leq d_T]\} \tag{9.18}$$

$$DIS(\mathcal{H}) = \left\{x \in X : \exists h, g \in \mathcal{H} \; s.t. \; h(x) \neq g(x) \cap \left[\left[(x - \overline{x_p})^T C^{-1} (x - \overline{x_p})\right] \leq d_T\right]\right\} \tag{9.19}$$

Compared to the conventional AL approaches, an advantage of the proposed method is that the disagreement region is limited right at the beginning by leveraging the knowledge of uncertainty propagated at the input side. This reduces the search space to optimize the AL heuristics, thereby improving the computational efficiency.

2. Active Learning over the sub-sampled querying space: Once the querying space is sub-sampled using the threshold (d_T) on Mahalanobis distance and the subsequent disagreement region is estimated using eq. (9.19), the next step is to perform AL. Consider a

set V of all candidate decision models. In AL, the unlabeled samples are iteratively processed in sequence, i.e., the labels Y_i of samples X_i in $DIS(V)$ are requested during each query. In this work, we employ uncertainty sampling query strategy [53, 81] to implement AL. It is important to note that the proposed approach can be extended to other AL query strategies as well. The set V is periodically updated by removing the decision models with relatively poor performance over the queried samples. We derive lower and upper bounds for label complexity in the context of AL approach considered in this work. For both lower and upper bounds, we first formulate the label complexity in no-noise case based on the work in [39]. These are further extended to noisy case, where we consider noise rate ν in eq. (2.17) to be analogous to uncertainty propagated from the model predictions back to the input side $\Delta \boldsymbol{x}_b$. The lower bounds presented in this work hold for any query-based learning algorithm, without involving any of the additional AL specific heuristics. Moreover, these lower bounds are based on the minimax analysis of label complexity, i.e., minimum label complexity corresponding to the worst case scenario. On the other hand, the computation of upper bounds on label complexity considers the AL-specific measures.

Theorem 3. *Given an AL algorithm that achieves a label complexity Λ, distribution $\mathcal{P}$ and any $\epsilon > 0$, there exists a distribution $\mathcal{P}_{XY}$ with marginal $\mathcal{P}$ over X, such that*

$$\Lambda(\epsilon, \delta, \mathcal{P}_{XY}) \geq \lceil \log_2((1-\delta)\mathcal{N}(2\epsilon, \mathcal{P})) \rceil \tag{9.20}$$

Proof. This proof uses the concept of ϵ-covering number of $\mathbb{C}$, indicated by $\mathcal{N}(\epsilon, \mathcal{P})$. For an arbitrary distribution $\mathcal{P}$, the ϵ-covering number of $\mathbb{C}$ is the smallest integer N, such that there exists a set of classifiers $\mathcal{H}$ with $|\mathcal{H}| = N$ for which $\bigcup_{h \in \mathcal{H}} B(h, \epsilon) = \mathbb{C}$ [39].

The proof is based on the fact of 2ϵ-packing of $\mathbb{C}$, i.e., there exists a set of classifiers $H \in \mathbb{C}$ with $|H| \geq \mathcal{N}(2\epsilon, \mathcal{P})$ such that every distinct $h, g \in H$ satisfy $\mathcal{P}(x : h(x) \geq g(x)) > 2\epsilon$. Assuming that the target function is chosen from H randomly, any learning algorithm that is capable of generating h with $E(h) \leq \epsilon$ can identify which $g \in H$ is the target classifier using:

$$g = \arg\min_{f \in H} \mathcal{P}(x : h(x) \neq f(x)) \tag{9.21}$$

The entropy of determining the target $g \in H$ is at least $\log_2(\mathcal{N}(2\epsilon, \mathcal{P}))$. If the output of the classifier is assumed to be binary, this is fundamentally the source of lower bound. The factor $(1 - \delta)$ accounts for the failure of the learning algorithm with a probability of δ. $\square$

Theorem 4. *Given an AL algorithm that achieves a label complexity Λ, sufficiently small $\epsilon, \delta > 0$, any $\Delta \boldsymbol{x}_b \in (0, 0.5)$, and if $\mathbb{C} \geq 3$, there exists a universal constant $q \in (0, \infty)$ and a distribution $\mathcal{P}_{XY}$ with $E(f^*) = \Delta \boldsymbol{x}_b$ such that*

$$\Lambda(\Delta \boldsymbol{x}_b + \epsilon, \delta, \mathcal{P}_{XY}) \geq q \left(\frac{\Delta \boldsymbol{x}_b^2}{\epsilon^2} \right) \left(d + Log \left(\frac{1}{\delta} \right) \right) \tag{9.22}$$

where d is the Vapnik-Chervonenkis dimension [237] and $Log(x) = \max\{\ln(x), 1\}$

Proof. The detailed proof of this Theorem is adapted from that of Theorem 4.3 in [39]. For the sake of brevity, we provide a brief sketch of proof below:

- The proof is pursued in two parts: the part corresponding to the d term and then for the $Log\left(\frac{1}{\delta}\right)$ term.
- For d term:
  - When $d = 1$, the part corresponding to d term is superfluous and this step may be skipped.
  - When $d \geq 2$,
    - Define $\mathcal{P}(\{x_i\})$ for each $i \in \{1, 2, ..., (d-1)\}$
    - Assume a classifier $\hat{h}_{nt}$ generated by the learning algorithm that achieves label complexity Λ, using budget $n \in \mathbb{N}$, $t = \{t_i\}_{i=1}^{d-1} \in \{0, 1\}^{d-1}$
    - Compute the expression for probability of $er(\hat{h}_{nt}) - er(f^*)$
    - Using law of total probability, find the conditions on $t = \{t_i\}_{i=1}^{d-1}$ during which the expression evaluated above holds true

$*$ Find the condition on δ that gives $\Lambda(\Delta \boldsymbol{x}_b + \epsilon, \delta, \mathcal{P}_{XY}) > n$

- For $Log\left(\frac{1}{\delta}\right)$ term: Similar steps as d terms when $d \geq 2$.

□

Next, the upper bounds on the label complexity of AL algorithm are formulated. In order to implement AL, a set V of all candidate decision models is considered. Then, the unlabeled samples are iteratively processed in a sequence by requesting labels Y_i for samples X_i in $DIS(V)$ during each query. Consequently, the set V is also updated during each iteration in AL. The AL algorithm with budget n is summarized below.

1. $m \leftarrow 0$, $t \leftarrow 0$, $V \leftarrow \mathbb{C}$
2. While $t < n$ and $m < 2^n$,
   (a) $m \leftarrow m + 1$
   (b) If $X_m \in DIS(V)$
      i. Request label Y_m
      ii. Let $V \leftarrow \{h \in V : h(X_m) = Y_m\}$
      iii. $t \leftarrow t + 1$
3. Return any $h \in V$

Here, the label complexity analysis focuses on bounding the total number of label requests made by the AL algorithm, which is sufficient to guarantee that $\forall h \in V$, $E(h) \leq \epsilon$ with at least $(1 - \delta)$ probability. To help in these bounds derivation, the following Lemma related to general concentration inequalities is introduced [39, 238]:

Lemma 2. *For any* $\gamma \in (0, 1)$, *any* $m \in \mathbb{N}$, *assuming* $\epsilon_m = \left(\frac{ad}{m}\right)^{\frac{1}{2-\alpha}}$, *and*

$$U(m, \gamma) = c \min \begin{cases} \left(\frac{a(dLog(\theta(a\epsilon_m^{\alpha})) + Log(1/\gamma))}{m}\right)^{\frac{1}{2-\alpha}} \\ \frac{dLog(\theta(d/m)) + Log(1/\gamma)}{m} + \sqrt{\frac{\nu(dLog(\theta(\nu)) + Log(1/\gamma))}{m}} \end{cases}$$

with probability at least $1-\gamma$, $\forall h \in \mathbb{C}$, *there exists a universal constant* $c \in (1,\infty)$ *such that the following inequalities hold:*

$$E(h) - E(f^*) \leq \max\{2(E_m(h) - E_m(f^*)), U(m,\gamma)\},$$

$$E_m(h) - \min_{g\in\mathbb{C}} E_m(g) \leq \max\{2(E(h) - E(f^*)), U(m,\gamma)\}$$

Simplifying further using algebra, the following also holds for any $m \in \mathbb{N}$, $c \in [1,\infty)$ *and* $\epsilon, \gamma \in (0,1)$*:*

$$m \geq c' a \epsilon^{\alpha-2}(dLog(\theta(a\epsilon^{\alpha})) + Log(1/\gamma)) \implies U(m,\gamma) \leq \epsilon$$

$$m \geq c' \left(\frac{\nu+\epsilon}{\epsilon^2}\right)(dLog(\theta(\nu+\epsilon)) + Log(1/\gamma)) \implies U(m,\gamma) \leq \epsilon$$

Theorem 5. *The AL algorithm achieves a label complexity* Λ *such that for any distribution* $\mathcal{P}_{XY}$, $\forall \epsilon, \delta \in (0,1)$,

$$\Lambda(\epsilon, \delta, \mathcal{P}_{XY}) \leq \theta(\epsilon)\left(Log\left(\frac{Log(1/\epsilon)}{\delta}\right)\right)\left(Log\left(\frac{d_T}{\epsilon||\overline{X_S}||}\right)\right) \tag{9.23}$$

where d_T *is the threshold on Mahalanobis distance and* $||\overline{X_S}||$ *is the* L_2*-norm of mean of the feature vectors in the labeled set.*

Proof. Fix any $\epsilon, \delta \in (0,1)$ and the budget of AL algorithm $n \in \mathbb{N}$ that satisfies

$$n \geq \log_2\left(\frac{2}{\delta}\right) 8ec'\theta(\epsilon)\left(dLog(\theta(\epsilon)) + 2Log\left(2\log_2(4/\epsilon)/\delta\right)\right)\log_2\left(\frac{2d_T}{\epsilon||\overline{X_S}||}\right)$$

for some universal constant $c' \in [1,\infty)$. Consider $M \subseteq \{0,...,2^n\}$ to be the values of m obtained during the querying process in AL. For every $m \in M$, let V_m be the value of V with corresponding value of m. For an event A of probability 1, any $m \in \mathbb{N}$ has $Y_m = f^*(X_m)$.

$\implies$ By induction, $\forall m \in M$, $f^* \in V_m = V_m^* = \{h \in \mathbb{C} : E_m(h) = 0\}$

Assume $i_\epsilon = \lceil \log_2 \left(\frac{1}{\epsilon}\right) \rceil$ and let $I = \{0, ..., i_\epsilon\}$. For $i \in I$, let $\epsilon_i = 2^{-i}$ and $m_0 = 0$; $\forall i \in I \backslash \{0\}$, let

$$m_i = \left\lceil c' \left(\frac{d_T}{\epsilon_i ||\overline{X_S}||} \right) \left(dLog(\theta(\epsilon_i)) + Log \left(\frac{2(2 + i_\epsilon - i)^2}{\delta} \right) \right) \right\rceil$$

Using union bound and Lemma 2, it can be deduced that every $i \in I$ has

$$sup_{h \in V^*_{m_i}} er(h) \leq \epsilon_i \tag{9.24}$$

for any event A_δ with probability at least $1 - \sum_{i=1}^{i_\epsilon} \frac{\delta}{2(2+i_\epsilon - i)^2} > 1 - \frac{\delta}{2}$

Considering the event A, while $m \leq m_{i_\epsilon}$, no. of label requests made by AL algorithm can be written as

$$\sum_{m=1}^{\min\{m_{i_\epsilon}, \max M\}} \mathbb{1}_{DIS(V_{m-1})}(X_m) = \sum_{m=1}^{\min\{m_{i_\epsilon}, \max M\}} \mathbb{1}_{DIS(V^*_{m-1})}(X_m)$$

The maximum value of this term is

$$\sum_{m=1}^{m_{i_\epsilon}} \mathbb{1}_{DIS(V^*_{m-1})}(X_m) = \sum_{i \in I \backslash \{0\}} \sum_{m=m_{i-1}+1}^{m_i} \mathbb{1}_{DIS(V^*_{m-1})}(X_m) \tag{9.25}$$

V^*_m is monotonous in m. Thus, for $m \in \{m_{i-1} + 1, ..., m_i\}$ and any $i \in I \backslash \{0\}$, we have $DIS(V^*_{m-1}) \subseteq DIS(V^*_{m_{i-1}})$. Using eq. (9.24), $V^*_{m-1} \subseteq B(f^*, \epsilon_{i-1})$ for any event E_δ. This means $DIS(V^*_{m_{i-1}}) \subseteq DIS(B(f^*, \epsilon_{i-1}))$. So, the maximum value of (9.25) is:

$$\sum_{i \in I \backslash \{0\}} \sum_{m=m_{i-1}+1}^{m_i} \mathbb{1}_{DIS(B(f^*, \epsilon_{i-1}))}(X_m) \tag{9.26}$$

which is essentially the sum of m_{i_ϵ} independent Bernoulli random variables with an expected value of

$\sum_{i \in I \backslash \{0\}} (m_i - m_{i-1}) \mathcal{P}(DIS(B(f^*, \epsilon_{i-1})))$. Using Chernoff bound, the maximum value of (9.26) for an event A'_δ with probability at least $1 - \frac{\delta}{2}$ is

$$\log_2\left(\frac{2}{\delta}\right) + 2e \sum_{i \in I\backslash\{0\}} (m_i - m_{i-1}) \mathcal{P}(DIS(B(f^*, \epsilon_{i-1}))) \tag{9.27}$$

It is known that $\mathcal{P}(DIS(B(f^*, \epsilon_{i-1}))) \leq \theta(\epsilon_{i-1})\epsilon_{i-1}$ (definition of disagreement coefficient). Hence, $\forall i \in I\backslash\{0\}$, the maximum value of $m_i \mathcal{P}(DIS(B(f^*, \epsilon_{i-1})))$ is:

$$4c'\theta(\epsilon_{i-1}) \left(dLog(\theta(\epsilon_i)) + Log\left(\frac{2(2 + i_\epsilon - i)^2}{\delta}\right)\right) \tag{9.28}$$

For $i \in I\backslash\{0\}$, $\theta(\epsilon_{i-1}) \leq \theta(\epsilon_i) \leq \theta(\epsilon)$

$$\implies (9.27) < \log_2\left(\frac{2}{\delta}\right) 8ec'\theta(\epsilon) \left(dLog(\theta(\epsilon)) + 2Log\left(2\log_2(4/\epsilon)/\delta\right)\right) \log_2\left(\frac{2d_T}{\epsilon||\overline{X_S}||}\right) \leq n \tag{9.29}$$

It is proved that: (i) for an event $A \cap A_\delta \cap A'_\delta$, $sup_{h \in V^*_{m_{i_\epsilon}}} E(h) \leq \epsilon_{i_\epsilon} \leq \epsilon$; (ii) while $m \leq m_{i_\epsilon}$, no. of label requests made by AL algorithm is less than n. Also, $\max M \geq m_{i_\epsilon}$ ($\because 2^n > m_{i_\epsilon}$) $\implies \hat{h} \in V^*_{m_{i_\epsilon}} \implies E(\hat{h}) \leq \epsilon$. By using union bound, it can be noted that the probability of $A \cap A_\delta \cap A'_\delta$ is at least $(1 - \delta)$ and

$$\log_2\left(\tfrac{2}{\delta}\right) 8ec'\theta(\epsilon) \left(dLog(\theta(\epsilon)) + 2Log\left(2\log_2(4/\epsilon)/\delta\right)\right) \log_2\left(\tfrac{2d_T}{\epsilon||\overline{X_S}||}\right) \leq$$
$$\theta(\epsilon) \left(Log\left(\tfrac{Log(1/\epsilon)}{\delta}\right)\right) \left(Log\left(\tfrac{d_T}{\epsilon||\overline{X_S}||}\right)\right)$$

□

Theorem 6. *The AL algorithm achieves a label complexity Λ such that for any distribution $\mathcal{P}_{XY}$, $\forall \epsilon, \delta \in (0, 1)$,*

$$\begin{gathered}\Lambda(\Delta\boldsymbol{x}_b + \epsilon, \delta, \mathcal{P}_{XY}) \leq \\ \theta(\Delta\boldsymbol{x}_b + \epsilon) \left(\tfrac{\Delta\boldsymbol{x}_b^2}{\epsilon^2} + Log\left(\tfrac{d_T}{\epsilon||\overline{X_S}||}\right)\right) \left(dLog(\theta(\Delta\boldsymbol{x}_b + \epsilon)) + Log\left(\tfrac{Log(1/\epsilon)}{\delta}\right)\right)\end{gathered} \tag{9.30}$$

Proof. Fix any $\epsilon, \delta \in (0, 1)$ and the budget of AL algorithm $n \in \mathbb{N}$ that satisfies

$$n > c''\theta(\Delta \boldsymbol{x}_b + \epsilon)\left(\frac{\Delta \boldsymbol{x}_b^2}{\epsilon^2} + Log\left(\frac{d_T}{\epsilon||\overline{X_S}||}\right)\right)\left(dLog(\theta(\Delta \boldsymbol{x}_b + \epsilon)) + Log\left(\frac{Log(1/\epsilon)}{\delta}\right)\right)$$

where c'' is a numerical constant. Following the steps as elucidated in the Proof of Theorem 5, consider $M \subseteq \{0, ..., 2^n\}$ to be the values of m obtained during AL loop. For every $m \in M$, let V_m be the value of V with corresponding value of m. Using union bound and Lemma 2, it can be deduced that every $m \in \mathbb{N}$ with $\log_2(m) \in \mathbb{N}$ satisfies

$$E_m(f^*) - \min_{g \in \mathbb{C}} E_m(g) \leq U(m, \delta_m) \tag{9.31}$$

for any event A_δ with probability at least $1 - \sum_{i=1}^{\infty} \frac{\delta}{(1+i)^2} > 1 - \frac{2\delta}{3}$. Additionally, $\forall h \in \mathbb{C}$,

$$E(h) - E(f^*) \leq \max\{2(E_m(h) - E_m(f^*)), U(m, \delta_m)\} \tag{9.32}$$

Assume $i_\epsilon = \left\lceil \log_2\left(\frac{2}{\epsilon}\right)\right\rceil$ and let $I = \{0, ..., i_\epsilon\}$. For $i \in I$, let $\epsilon_i = 2^{-i}$. Denote $\lceil x \rceil_2 = 2^{\lceil \log_2(x) \rceil}$, i.e., smallest power of 2 that is at least as large as x. Assume $m_0 = 0$; $\forall i \in I \backslash \{0\}$, let

$$m_i' = 4c'\left(\frac{\Delta x_b}{\epsilon_i^2} + \frac{d_T}{\epsilon||\overline{X_S}||}\right)\left(dLog(\theta(\Delta x_b + \epsilon_i)) + Log\left(\frac{4\log_2(4c'/\epsilon_i)}{\delta}\right)\right)$$

with $m_i = \lceil m_i' \rceil_2$. Combining this with eq. (9.32), for an event A_δ, it can be said that

$$\forall h \in V_{m_i}, E(h) - E(f^*) \leq 2U(m_i, \delta_{m_i}) \ (\because f^* \in V_{m_i - 1})$$

With $\gamma = \delta_{m_i}$ in Lemma 2, $U(m_i, \delta_{m_i}) \leq \epsilon_i$ for every $i \in I \backslash \{0\}$. Moreover, $\forall h \in V_{m_0} = \mathbb{C}$, $E(h) - E(f^*) \leq 2\epsilon_0$ holds trivially. So, for an event E_δ, the following holds true for all $i \in I$ with $m_i \in M$:

$$\forall h \in V_{m_i}, E(h) - E(f^*) \leq 2\epsilon_i \tag{9.33}$$

$\implies$ It is sufficient to show that $m_{i_\epsilon} \in M$, in order to complete the proof.

While $m \leq m_{i_\epsilon}$, number of label requests made by AL algorithm can be written as

$$\sum_{m=1}^{\min\{m_{i_\epsilon}, \max M\}} \mathbb{1}_{DIS(V_{m-1})}(X_m) = \sum_{i=1}^{i_\epsilon} \sum_{m=m_{i-1}+1}^{\min\{m_i, \max M\}} \mathbb{1}_{DIS(V_{m-1})}(X_m)$$

From eq. (9.33), for $m \in \{m_{i-1}+1, ..., m_i\}$ and $i \in I \backslash \{0\}$, $DIS(V_{m-1}) \subseteq DIS(V_{m_{i-1}}) \subseteq DIS(\mathbb{C}(2\epsilon_{i-1}))$. So, the maximum value of no. of label requests made by AL algorithm is

$$\sum_{i=1}^{i_\epsilon} \sum_{m=m_{i-1}+1}^{m_i} \mathbb{1}_{DIS(\mathbb{C}(2\epsilon_{i-1}))}(X_m)$$

which is essentially the sum of independent Bernoulli random variables. Using Chernoff bound, its maximum value for an event A'_δ with probability at least $1 - \frac{\delta}{3}$ is

$$\log_2\left(\frac{3}{\delta}\right) + 2e\sum_{i=1}^{i_\epsilon}(m_i - m_{i-1})\mathcal{P}(DIS(\mathbb{C}(2\epsilon_{i-1}))) \tag{9.34}$$

Using triangle inequality, $\mathbb{C}(2\epsilon_{i-1}) \subseteq B(f^*, 2\Delta x_b + 2\epsilon_{i-1})$

$\implies$ $\mathcal{P}(DIS(\mathbb{C}(2\epsilon_{i-1}))) \leq \theta(2(\Delta x_b + \epsilon_{i-1}))2(\Delta x_b + \epsilon_{i-1})$ ($\because$ definition of disagreement coefficient). Hence, $\forall i \in I \backslash \{0\}$, the maximum value of $m_i\mathcal{P}(DIS(\mathbb{C}(2\epsilon_{i-1})))$ is:

$$\theta(2(\Delta x_b + \epsilon_{i-1}))\left(\frac{\Delta \boldsymbol{x}_b^2}{\epsilon_i^2} + 1\right)\left(dLog(\theta(\Delta \boldsymbol{x}_b + \epsilon_i)) + Log\left(\frac{Log(1/\epsilon_i)}{\delta}\right)\right) \tag{9.35}$$

Using properties of disagreement coefficient similar that in Proof of Theorem 5,

$$(9.34) \leq c''\theta(\Delta \boldsymbol{x}_b + \epsilon)\left(\frac{\Delta \boldsymbol{x}_b^2}{\epsilon^2} + Log\left(\frac{d_T}{\epsilon||\overline{X_S}||}\right)\right)\left(dLog(\theta(\Delta \boldsymbol{x}_b + \epsilon)) + Log\left(\frac{Log(1/\epsilon)}{\delta}\right)\right) < n \tag{9.36}$$

for a suitable choice of the constant c''.

Hence, it is proved that: (i) for an event $A_\delta \cap A'_\delta$, $E(\hat{h}) - E(f^*) \leq sup_{h \in V_{m_{i_\epsilon}}} E(h) - E(f^*) \leq \epsilon$ (ii) while $m \leq m_{i_\epsilon}$, number of label requests made by AL algorithm is less than n. Moreover,

by using union bound, it can be noted that the probability of $A_\delta \cap A'_\delta$ is at least $(1-\delta)$. $\square$

9.3 Experimental Results

The proposed methodology is evaluated across both classification and regression tasks. In case of classification, a pancreatic cancer dataset derived from [103] is considered. This encompasses the quantification of protease/arginase activity through the utilisation of highly sensitive nanobiosensors. It comprises a collection of eight biomarkers, namely arginase, matrix metalloproteinase-1, -3, and -9, cathepsin-B and -E, urokinase plasminogen activator, and neutrophil elastase. The identified biomarkers were collected from the NCBI Gene Expression Omnibus (GEO) database, that is available in public domain. The proteases that show notable variations in expression levels between two samples, namely a primary tumour sample and a sample of healthy tissue, were required to originate from Homo sapiens. Consequently, these proteases were classified as biomarkers. The methodology utilised for gene expression analysis involved gathering data from Gene Symbol, Entrez Gene ID, NCBI GEO, and Unigene ID. This data was then used to calculate the characteristics associated with a set of seven proteases and arginase for each sample in the dataset. The detection of pancreatic cancer is modeled as a hierarchical classification scheme with a binary classifier at each hierarchical step, as described in [115]. It comprises of 159 samples classified into the following categories: "Healthy", "Pancreatitis", "Localized" pancreatic cancer and "Metastatic" pancreatic cancer. "Healthy" denotes the data corresponding to healthy volunteers. "Pancreatitis" indicates the ones with inflammation in pancreas. While "Localized" pancreatic cancer samples indicate the absence of any signs that the disease has spread outside the pancreas, "Metastatic" pancreatic cancer implies that the cancer has spread to other parts of the body too. For regression, we consider medical insurance costs dataset from [239]. The objective of this dataset is to predict the insurance charges for people based on their age, gender, BMI, number of children, smoking status and region of residence. The sample size of medical insurance costs dataset is 1338.

A simple neural network architecture consisting of two linear layers with a softmax at

Table 9.1: Theoretical Results of the proposed uncertainty-guided AL framework for classification task

Sr. No.	Level of Uncertainty		Bounds on Label Complexity	
	Output Layer	Input Layer	Lower Bound	Upper Bound
1	0	0	69	82
2	10% (0.1)	0.24	73	87
3	20% (0.2)	0.36	79	95
4	30% (0.3)	0.49	86	103

Table 9.2: Theoretical Results of the proposed uncertainty-guided AL framework for regression task

Sr. No.	Level of Uncertainty		Bounds on Label Complexity	
	Output Layer	Input Layer	Lower Bound	Upper Bound
1	0	0	551	624
2	10% (0.1)	0.35	604	712
3	20% (0.2)	0.56	675	785
4	30% (0.3)	0.74	764	857

the end is employed for training the decision models. Uncertainty sampling query strategy is used for strategic selection of informative samples within the iterative querying process of AL environment. The initial labeled datasets are curated by randomly selecting 10% of the total instances in the respective datasets. Subsequently, this data is utilised to train an initial neural network-based decision model. After training an initial model, the querying procedure is commenced and the decision models are updated after each query step. In both classification and regression tasks, theoretical as well as empirical results are presented. For theoretical results, different thresholds for uncertainties in model predictions are considered and propagated back to the input side, as elucidated in Section 9.2.1. Uncertainty in model predictions is quantified by variations in the output of softmax layer for the classification task, and that of final neural network layer for the regression task. Further, the lower and upper bounds on label complexity are estimated, as discussed in Section 9.2.2. These are reported in Table 9.1 and Table 9.2 for classification and regression tasks, respectively. It can be seen that if the level of uncertainty is higher, the bounds on label complexity increase. This is intuitive because it requires more samples/queries for the AL algorithm to learn an efficient decision model, in case of higher uncertainties at input/output.

The empirical evaluation is also performed by actually implementing AL over both clas-

Table 9.3: Empirical Results of the proposed uncertainty-guided AL framework for classification task

Sr. No.	Uncertainty at Output Layer	Initial Classification Accuracy	Classification Accuracy after specific no. of queries				Remarks
			20	40	60	80	
1	0	51.78%	63.71%	71.32%	85.35%	95.31%	
2	10% (0.1)	44.84%	56.94%	65.17%	83.52%	90.92%	$\sim$ 95% with 85 queries
3	20% (0.2)	39.14%	48.43%	59.56%	72.39%	82.13%	$\sim$ 95% with 93 queries
4	30% (0.3)	33.41%	45.67%	54.28%	66.37%	75.63%	$\sim$ 95% with 100 queries

Table 9.4: Empirical Results of the proposed uncertainty-guided AL framework for regression task

Sr. No.	Uncertainty at Output Layer	Initial Predictive Performance	Predictive Performance after specific no. of queries				Remarks
			150	300	450	600	
1	0	54.10%	65.93%	72.59%	87.07%	95.74%	
2	10% (0.1)	49.15%	62.32%	71.37%	80.14%	91.11%	$\sim$ 95% with 695 queries
3	20% (0.2)	42.74%	59.23%	65.06%	76.13%	84.74%	$\sim$ 95% with 760 queries
4	30% (0.3)	35.49%	46.16%	59.36%	69.62%	80.45%	$\sim$ 95% with 848 queries

sification and regression tasks in order to check if a desirable performance of decision model is achieved within the label complexity bounds established by theoretical results. The empirical results are presented in Table 9.3 and Table 9.4 for classification and regression tasks, respectively. It can be clearly seen that $\sim$ 95% classification accuracy (or predictive performance, for regression task) is achieved by no. of queries that are within the upper bounds estimated by theoretical results. This holds true for both classification as well as regression tasks and demonstrates the efficacy of the proposed approach. At a specific number of queries, the classification accuracy reduces with increase in the level of uncertainty. This is again intuitive as AL algorithm needs higher number of queries when input and output uncertainty increases.

Knowing the label complexity bounds, longitudinal analyses (averaged across specific number of samples) are carried out by considering the input uncertainty levels estimated by theoretical results. This is to validate the performance level of decision models at lower

Table 9.5: Longitudinal Analyses of the proposed uncertainty-guided AL framework for classification task (averaged across 25 samples)

Sr. No.	Level of Uncertainty		Bounds on Label Complexity		Classification Accuracy at	
	Input Layer	Output Layer	Lower Bound	Upper Bound	Lower Bound	Upper Bound
1	0	0	69	82	89.66%	96.12%
2	0.24	0.1	73	87	86.43%	95.96%
3	0.36	0.2	79	95	82.03%	96.31%
4	0.49	0.3	86	103	79.58%	95.84%

Table 9.6: Longitudinal Analyses of the proposed uncertainty-guided AL framework for regression task (averaged across 200 samples)

Sr. No.	Level of Uncertainty		Bounds on Label Complexity		Predictive Performance at	
	Input Layer	Output Layer	Lower Bound	Upper Bound	Lower Bound	Upper Bound
1	0	0	551	624	92.35%	96.05%
2	0.35	0.1	604	712	91.67%	95.56%
3	0.56	0.2	675	785	88.61%	96.19%
4	0.74	0.3	764	857	87.92%	96.41%

and upper bounds of label complexity and demonstrated in Table 9.5 and Table 9.6 for classification and regression tasks, respectively. The bounds of label complexity established by theoretical results are successfully verified for all the experimental scenarios.

Furthermore, uncertainty calibration is also evaluated for all the experiments performed. Negative Log Likelihood (NLL) [240] and Brier Score (BS) [241] are metrics used to indirectly quantify uncertainty calibration, as shown in Table 9.7 and Table 9.8 for classification and regression tasks, respectively. The values of NLL are observed to increase with an increase in level of uncertainty, thereby demonstrating the high-quality uncertainty calibration of decision models trained using the proposed approach. A similar trend is witnessed for BS values as well.

Table 9.7: Evaluating uncertainty calibration of the proposed uncertainty-guided AL framework for classification task

Sr. No.	Level of Uncertainty		Negative Log Likelihood	Brier Score
	Input Layer	Output Layer		
1	0	0	-	-
2	0.24	0.1	-0.82	0.05
3	0.36	0.2	-0.71	0.07
4	0.49	0.3	-0.54	0.08

Table 9.8: Evaluating uncertainty calibration of the proposed uncertainty-guided AL framework for regression task

Sr. No.	Level of Uncertainty		Negative Log Likelihood	Brier Score
	Input Layer	Output Layer		
1	0	0	-	-
2	0.35	0.1	-0.76	0.03
3	0.56	0.2	-0.53	0.04
4	0.74	0.3	-0.44	0.06

9.4 Summary

In this chapter, a novel framework is proposed to leverage the knowledge of input and output uncertainties in NN-based decision models to guide querying process in the context of AL. A tolerable threshold on the uncertainty of NN predictions is considered and propagated back to input side using the ADF approach. Then, this information is used to guide informative sample selection in AL. The lower and upper bounds for label complexity are also derived in the context of AL approach considered in this work. It is worthwhile to note that the proposed methodology is generic and hence, agnostic to AL setup, query strategy and the base decision model. Experimental results demonstrate the applicability of the proposed framework across both classification and regression tasks. Empirical evaluation is performed by actually implementing AL in order to check if a desirable performance of decision model is achieved within the label complexity bounds established by theoretical results. In order to validate the performance level of decision models at theoretical label complexity bounds, longitudinal analyses is carried out by considering the input uncertainty levels estimated by theoretical results. The next chapter concludes the thesis and highlights the potential future research directions.

Chapter 10

Conclusions and Future Work

This chapter provides a summary of the contributions of this dissertation and outlines potential directions for future research.

10.1 Conclusions

This dissertations addresses fundamental questions related to learning and inferencing challenges in HITL decision systems. The primary motive is to develop frameworks that support building an efficient decision support tool for any HITL environment. The decision-making process facilitated by this support tool should prioritize interpretability and resilience against modeling errors, measurement noise, and biases inherent in both human and AI components. Such integrated frameworks are particularly valuable for safety-critical applications, where extensive human oversight is necessary, and errors introduced by either human or AI components can have costly repercussions. For instance, a robust and transparent decision support tool in medical diagnosis can assist doctors and clinicians in considering the need for additional tests, thereby enabling more informed decision-making with increased confidence. The accomplishments of this dissertation can be summarized as follows.

Chapter 3 introduces a hybrid query strategy-based framework, which addresses the practical challenges in AL, namely, cold-start, oracle uncertainty and evaluating model per-

formance in the absence of ground truth. The robustness of the proposed framework is evaluated across three datasets from different environments and industrial settings. The merits of using a pre-clustering approach to address the cold-start problem is exemplified by experiments performed on the process data collected from a simulated ethanol plant. The utility of hybrid query strategies is established with the help of experiments performed on the drug consumption dataset. It is concluded that the use of hybrid query strategies helps in reducing the computational cost, while delivering at par or better performance than the individual query strategies. The contribution of all the elements together in the proposed framework is then demonstrated through experiments carried out on the rolling element bearing test data from Case Western Reserve University.

The idea of AL is further extended to representation learning in non-Euclidean space like graphs in Chapter 4. An AL framework is presented to address node classification task in attributed graphs using a convex optimization approach. The proposed method integrates both topological information as well as node attributes within the learning scheme. The decision models are trained and updated using GraphSAGE, an inductive node embedding approach that learns both the topological structure and distribution of features for a node in its local neighborhood. Graph theoretic measures are used as AL heuristics to select the most informative nodes in the graph for annotation. It is observed that betweenness centrality and EGR consistently outperform the baseline and improve the classification performance by upto 10%.

In Chapter 5, the impact of seven behavioral biases, namely, herding bias, cognitive bias, hot-hand fallacy, representative bias, anchoring bias, gambler's fallacy and regret-aversion bias is illustrated using experiments conducted on a real-world pancreatic cancer dataset. AL is represented in the form of a collaborative decision environment of AI engines and human experts, and the annotation by human experts is formulated as a two-step process. Further, the behavioral biases are simulated by dispensing pre-manipulated user inputs based on the foundational understanding of respective biases during the iterative query steps. Finally, the impacts of these biases on the performance of AL strategies are assessed by comparing classification accuracy score of the decision model against a reference case, which does not

assimilate any sort of behavioral bias. It is observed that the performance deteriorates significantly when the human decisions are influenced by each of the behavioral biases. Chapter 6 establishes a systematic framework for modeling, tracking and adaptation of behavioral biases in a collaborative human-AI decision setup is presented. The human-AI interactive decision environment is modled in the form of a coupled dynamical system (using communication graphs) that evolves as more queries are made in the context of AL frameworks. A novel ESIS strategy is proposed for tracking behavioral biases within AL settings. A model adaptation framework is introduced to handle the tracked biases. This is executed by allocating suitable weights to the training samples during the iterative querying process. The proposed framework is evaluated by conducting experiments on real-world pancreatic cancer dataset.

In Chapter 7, two case studies are presented to demonstrate ways of incorporating observational biases within the learning frameworks. The first case study illustrates a GNN-based framework for identification of potential biomarkers for biological phenomena. It involves incorporating knowledge from an existing database to develop a graph that incorporates protein-protein interactions as edge connections and is used as input data to identify the potential biomarkers. The second case study proposes a novel active foundational model for fault diagnosis of the electrical machines by combining aspects of AL and contrastive SSL during backbone training. The proposed framework consists of two major components: (i) constructing the backbone fault diagnosis model, and (ii) fine-tuning based on specific requirements of various target tasks. Chapter 8 exemplifies two case studies to demonstrate different ways of explaining the predictions made by decision models. The first case study focuses on early-stage detection of PC. The use of ultrasensitive nanobiosensors for protease/arginase detection is combined with information fusion based statistical framework to detect PC at the localized stage by means of a simple Liquid Biopsy. The proposed hierarchical SDS framework provides confidences associated with the predicted labels in the form of probability values for each class, which makes the decisions more "interpretable". This information can be used to clinicians in order to perceive the lack of confidence in model predictions and to examine if any further tests are required before making a final decision.

The second case study on early-stage detection of OC incorporates feature relevance technique to explain the predictions of corresponding decision models. This provides a feature relevance score for all the features corresponding to a sample under consideration.

Finally, Chapter 9 devises a novel framework to leverage the knowledge of input and output uncertainties in NN-based decision models to guide querying process in the context of AL. A tolerable threshold on the uncertainty of NN predictions is considered and propagated back to input side using the ADF approach. Then, this information is used to guide informative sample selection in AL. The lower and upper bounds for label complexity are also derived in the context of AL approach considered in this work. It is worthwhile to note that the proposed methodology is generic and hence, agnostic to AL setup, query strategy and the base decision model. Experimental results demonstrate the applicability of the proposed framework across both classification and regression tasks. Empirical evaluation is performed by actually implementing AL in order to check if a desirable performance of decision model is achieved within the label complexity bounds established by theoretical results.

10.2 Future Work

This section outlines potential future research directions related to HITL decision systems. The possible extensions of the work presented in this dissertation are enumerated below:

- Chapter 3 presents a hybrid query strategy-based framework, which addresses the practical challenges in AL, namely, cold-start, oracle uncertainty and evaluating model performance in the absence of ground truth. However, there are other practical challenges in AL, such as variable labeling costs and changing model classes. Developing AL frameworks to address these challenges can be pursued as a future extension of this work. The integration of trust aspects for the model as well as human annotator in the context of XAI is another potential future research direction.

- Chapter 4 illustrates an AL framework to address node classification task in attributed graphs using a convex optimization approach. The analysis reveals that betweenness

centrality and EGR consistently outperform the baseline method, leading to enhanced classification performance. However, there are many other graph theoretic measures which can potentially further improve model performance as well as robustness. Future extensions of this work include exploring the utility of other graph theoretic measures and evaluating the proposed framework over larger graphs as well as graphs from different application domains.

- Chapter 5 studies the impact of seven behavioral biases, namely, herding bias, cognitive bias, hot-hand fallacy, representative bias, anchoring bias, gambler's fallacy and regret-aversion bias on AL strategies. The future extensions of this work include ways to incorporate algorithmic biases and implement the corresponding framework across datasets from different domains.

- Chapter 6 demonstrates a systematic framework for modeling, tracking and adaptation of behavioral biases in a collaborative human-AI decision setup. The future work include ways to detect and handle algorithmic biases as well, within the learning scheme. This would lead to the development of a framework that addresses anomalies related to both human as well as as AI components of the collaborative decision environment. The performance of proposed framework also needs to be evaluated across data from different application domains.

- The human-AI collaborative decision setup naturally appears as a Reinforcement Learning (RL) environment with human expert and AI engine as the decision agents. This setup can be viewed as a two-player cooperative game, where human expert and AI engine work together to achieve a common objective, i.e., to learn a robust decision model. Such a synergistic framework allows learning of a generalized policy to sequentially select the instances for annotation my the human expert, as opposed to specific heuristic-based instance selection approach in AL settings. The possible future extension involves formulating the human-AI collaborative decision setup and associated biases as an RL problem, implementing it using appropriate RL methodology and

comparing its performance with AL counterpart.

- Chapter 7 presents two case studies are presented to demonstrate ways of incorporating observational biases within the learning frameworks. The future extension of the work on active foundational models involve extensive testing across more target tasks and incorporating physics information within the foundational models.

- Chapter 8 exemplifies two case studies to demonstrate different ways of explaining the predictions made by decision models. The future extension of this work involves exploring and analyzing other groups of explainability methodologies like pre-modeling, interpretable model and post-modeling approaches.

- Chapter 9 devises a novel framework to leverage the knowledge of input and output uncertainties in NN-based decision models to guide querying process in the context of AL. This work is limited to linear activation functions in neural networks. As part of the future work, this framework can be extended across different activation functions as well as representation learning in non-Euclidean space like graphs.

Bibliography

[1] Obdulia Covarrubias-Zambrano, Massoud Motamedi, Bill T Ameredes, Bing Tian, William J Calhoun, Yingxin Zhao, Allan R Brasier, Madumali Kalubowilage, Aruni P Malalasekera, Asanka S Yapa, et al. Optical biosensing of markers of mucosal inflammation. *Nanomedicine: Nanotechnology, Biology and Medicine*, page 102476, 2021.

[2] Robert Munro Monarch. *Human-in-the-Loop Machine Learning: Active learning and annotation for human-centered AI*. Simon and Schuster, 2021.

[3] Xingjiao Wu, Luwei Xiao, Yixuan Sun, Junhang Zhang, Tianlong Ma, and Liang He. A survey of human-in-the-loop for machine learning. *Future Generation Computer Systems*, 135:364–381, 2022.

[4] Shuji Hao, Peiying Hu, Peilin Zhao, Steven CH Hoi, and Chunyan Miao. Online active learning with expert advice. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 12(5):1–22, 2018.

[5] Ninareh Mehrabi, Fred Morstatter, Nripsuta Saxena, Kristina Lerman, and Aram Galstyan. A survey on bias and fairness in machine learning. *ACM Computing Surveys (CSUR)*, 54(6):1–35, 2021.

[6] Marius Protte, Rene Fahr, and Daniel E Quevedo. Behavioral economics for human-in-the-loop control systems design: Overconfidence and the hot hand fallacy. *IEEE Control Systems Magazine*, 40(6):57–76, 2020.

[7] Dang Minh, H Xiang Wang, Y Fen Li, and Tan N Nguyen. Explainable artificial intelligence: a comprehensive review. *Artificial Intelligence Review*, 55(5):3503–3568, 2022.

[8] Laurent Valentin Jospin, Wray Buntine, Farid Boussaid, Hamid Laga, and Mohammed Bennamoun. Hands-on bayesian neural networks–a tutorial for deep learning users. *arXiv preprint arXiv:2007.06823*, 2020.

[9] Alberto Bemporad. Active learning for regression by inverse distance weighting. *Information Sciences*, 2023.

[10] Deepesh Agarwal, Obdulia Covarrubias-Zambrano, Stefan Bossmann, and Balasubramaniam Natarajan. Impacts of behavioral biases on active learning strategies. In *2022 International Conference on Artificial Intelligence in Information and Communication (ICAIIC)*, pages 256–261. IEEE, 2022.

[11] Keyulu Xu, Chengtao Li, Yonglong Tian, Tomohiro Sonobe, Ken-ichi Kawarabayashi, and Stefanie Jegelka. Representation learning on graphs with jumping knowledge networks. In *International Conference on Machine Learning*, pages 5453–5462. PMLR, 2018.

[12] Justin Gilmer, Samuel S Schoenholz, Patrick F Riley, Oriol Vinyals, and George E Dahl. Neural message passing for quantum chemistry. In *International conference on machine learning*, pages 1263–1272. PMLR, 2017.

[13] Franco Scarselli, Marco Gori, Ah Chung Tsoi, Markus Hagenbuchner, and Gabriele Monfardini. The graph neural network model. *IEEE transactions on neural networks*, 20(1):61–80, 2008.

[14] Peter Battaglia, Razvan Pascanu, Matthew Lai, Danilo Jimenez Rezende, et al. Interaction networks for learning about objects, relations and physics. *Advances in neural information processing systems*, 29, 2016.

[15] Michaël Defferrard, Xavier Bresson, and Pierre Vandergheynst. Convolutional neural networks on graphs with fast localized spectral filtering. *Advances in neural information processing systems*, 29, 2016.

[16] David K Duvenaud, Dougal Maclaurin, Jorge Iparraguirre, Rafael Bombarell, Timothy Hirzel, Alán Aspuru-Guzik, and Ryan P Adams. Convolutional networks on graphs for learning molecular fingerprints. *Advances in neural information processing systems*, 28, 2015.

[17] Will Hamilton, Zhitao Ying, and Jure Leskovec. Inductive representation learning on large graphs. *Advances in neural information processing systems*, 30, 2017.

[18] Steven Kearnes, Kevin McCloskey, Marc Berndl, Vijay Pande, and Patrick Riley. Molecular graph convolutions: moving beyond fingerprints. *Journal of computer-aided molecular design*, 30(8):595–608, 2016.

[19] Thomas N Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. *International Conference on Learning Representations (ICLR)*, 2017.

[20] Yujia Li, Daniel Tarlow, Marc Brockschmidt, and Richard Zemel. Gated graph sequence neural networks. *International Conference on Learning Representations (ICLR)*, 2016.

[21] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, and Yoshua Bengio. Graph attention networks. *International Conference on Learning Representations (ICLR)*, 2018.

[22] Adam Santoro, David Raposo, David G Barrett, Mateusz Malinowski, Razvan Pascanu, Peter Battaglia, and Timothy Lillicrap. A simple neural network module for relational reasoning. *Advances in neural information processing systems*, 30, 2017.

[23] David Barrett, Felix Hill, Adam Santoro, Ari Morcos, and Timothy Lillicrap. Measuring abstract reasoning in neural networks. In *International conference on machine learning*, pages 511–520. PMLR, 2018.

[24] Zhitao Ying, Jiaxuan You, Christopher Morris, Xiang Ren, Will Hamilton, and Jure Leskovec. Hierarchical graph representation learning with differentiable pooling. *Advances in neural information processing systems*, 31, 2018.

[25] Muhan Zhang, Zhicheng Cui, Marion Neumann, and Yixin Chen. An end-to-end deep learning architecture for graph classification. In *Thirty-second AAAI conference on artificial intelligence*, 2018.

[26] Jiaxuan You, Rex Ying, and Jure Leskovec. Position-aware graph neural networks. In *International Conference on Machine Learning*, pages 7134–7143. PMLR, 2019.

[27] Jiaxuan You, Jonathan Gomes-Selman, Rex Ying, and Jure Leskovec. Identity-aware graph neural networks. *arXiv preprint arXiv:2101.10320*, 2021.

[28] Ehsan Elhamifar, Guillermo Sapiro, Allen Yang, and S Shankar Sasrty. A convex optimization framework for active learning. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 209–216, 2013.

[29] Ehsan Elhamifar, Guillermo Sapiro, and Rene Vidal. Finding exemplars from pairwise dissimilarities via simultaneous sparse recovery. *Advances in Neural Information Processing Systems*, 25, 2012.

[30] Ehsan Elhamifar, Guillermo Sapiro, and S Shankar Sastry. Dissimilarity-based sparse subset selection. *IEEE transactions on pattern analysis and machine intelligence*, 38 (11):2182–2197, 2015.

[31] Anurag Choudhary, Deepam Goyal, Sudha Letha Shimi, and Aparna Akula. Condition monitoring and fault diagnosis of induction motors: A review. *Archives of Computational Methods in Engineering*, 26:1221–1238, 2019.

[32] Meng-Hao Guo, Tian-Xing Xu, Jiang-Jiang Liu, Zheng-Ning Liu, Peng-Tao Jiang, Tai-Jiang Mu, Song-Hai Zhang, Ralph R Martin, Ming-Ming Cheng, and Shi-Min Hu. Attention mechanisms in computer vision: A survey. *Computational visual media*, 8 (3):331–368, 2022.

[33] Qingsong Wen, Tian Zhou, Chaoli Zhang, Weiqi Chen, Ziqing Ma, Junchi Yan, and Liang Sun. Transformers in time series: A survey. *arXiv preprint arXiv:2202.07125*, 2022.

[34] Jian Cen, Zhuohong Yang, Yinbo Wu, Xueliang Hu, Liwei Jiang, Honghua Chen, and Weiwei Si. A mask self-supervised learning-based transformer for bearing fault diagnosis with limited labeled samples. *IEEE Sensors Journal*, 2023.

[35] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PMLR, 2020.

[36] Rishi Bommasani, Drew A Hudson, Ehsan Adeli, Russ Altman, Simran Arora, Sydney von Arx, Michael S Bernstein, Jeannette Bohg, Antoine Bosselut, Emma Brunskill, et al. On the opportunities and risks of foundation models. *arXiv preprint arXiv:2108.07258*, 2021.

[37] 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.

[38] Sriram Anbalagan, Deepesh Agarwal, Balasubramaniam Natarajan, and Babji Srinivasan. Foundational models for fault diagnosis of electrical motors. In *2023 IEEE International Conference on Power Electronics, Smart Grid, and Renewable Energy (PESGRE)*, pages 1–6. IEEE, 2023.

[39] Steve Hanneke et al. Theory of disagreement-based active learning. *Foundations and Trends® in Machine Learning*, 7(2-3):131–309, 2014.

[40] Xiaojin Jerry Zhu. Semi-supervised learning literature survey. In *Technical report, Computer Sciences*. University of Wisconsin-Madison Department of Computer Sciences, 2005.

[41] Shuji Hao, Peiying Hu, Peilin Zhao, Steven CH Hoi, and Chunyan Miao. Online active learning with expert advice. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 12(5):1–22, 2018.

[42] Burr Settles. From theories to queries: Active learning in practice. In *Active Learning and Experimental Design workshop In conjunction with AISTATS 2010*, pages 1–18, 2011.

[43] Jingyu Shao, Qing Wang, and Fangbing Liu. Learning to sample: an active learning framework. In *2019 IEEE International Conference on Data Mining (ICDM)*, pages 538–547. IEEE, 2019.

[44] Wenbo Gong, Sebastian Tschiatschek, Richard Turner, Sebastian Nowozin, José Miguel Hernández-Lobato, and Cheng Zhang. Icebreaker: Element-wise active information acquisition with bayesian deep latent gaussian model. *arXiv preprint arXiv:1908.04537*, 2019.

[45] Anna Primpeli, Christian Bizer, and Margret Keuper. Unsupervised bootstrapping of active learning for entity resolution. In *European Semantic Web Conference*, pages 215–231. Springer, 2020.

[46] Yue Deng, KaWai Chen, Yilin Shen, and Hongxia Jin. Adversarial active learning for sequences labeling and generation. In *IJCAI*, pages 4012–4018, 2018.

[47] Christoph Sandrock, Marek Herde, Adrian Calma, Daniel Kottke, and Bernhard Sick. Combining self-reported confidences from uncertain annotators to improve label quality. In *2019 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8. IEEE, 2019.

[48] Adrian Calma and Bernhard Sick. Simulation of annotators for active learning: Uncertain oracles. In *IAL@ PKDD/ECML*, pages 49–58, 2017.

[49] Jinhua Song, Hao Wang, Yang Gao, and Bo An. Active learning with confidence-based answers for crowdsourcing labeling tasks. *Knowledge-Based Systems*, 159:244–258, 2018.

[50] Ramyar Saeedi, Keyvan Sasani, and Assefaw H Gebremedhin. Collaborative multi-expert active learning for mobile health monitoring: Architecture, algorithms, and evaluation. *Sensors*, 20(7):1932, 2020.

[51] Gaurav Gupta, Anit Kumar Sahu, and Wan-Yi Lin. Noisy batch active learning with deterministic annealing. *arXiv preprint arXiv:1909.12473*, 2019.

[52] Mohamed-Rafik Bouguelia, Slawomir Nowaczyk, KC Santosh, and Antanas Verikas. Agreeing to disagree: active learning with noisy labels without crowdsourcing. *International Journal of Machine Learning and Cybernetics*, 9(8):1307–1319, 2018.

[53] Deepesh Agarwal, Pravesh Srivastava, Sergio Martin-del Campo, Balasubramaniam Natarajan, and Babji Srinivasan. Addressing uncertainties within active learning for industrial iot. In *2021 IEEE 7th World Forum on Internet of Things (WF-IoT)*, pages 557–562. IEEE, 2021.

[54] Meng Fang and Xingquan Zhu. Active learning with uncertain labeling knowledge. *Pattern Recognition Letters*, 43:98–108, 2014.

[55] Guoliang He, Yifei Li, and Wen Zhao. An uncertainty and density based active semi-supervised learning scheme for positive unlabeled multivariate time series classification. *Knowledge-Based Systems*, 124:80–92, 2017.

[56] Rita Chattopadhyay, Zheng Wang, Wei Fan, Ian Davidson, Sethuraman Panchanathan, and Jieping Ye. Batch mode active sampling based on marginal probability distribution matching. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 7(3):1–25, 2013.

[57] Zheng Wang and Jieping Ye. Querying discriminative and representative samples for batch mode active learning. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 9(3):1–23, 2015.

[58] Jennifer Vandoni, Emanuel Aldea, and Sylvie Le Hégarat-Mascle. Evidential query-by-committee active learning for pedestrian detection in high-density crowds. *International Journal of Approximate Reasoning*, 104:166–184, 2019.

[59] Christopher Tosh and Sanjoy Dasgupta. Interactive structure learning with structural query-by-committee. In *Advances in Neural Information Processing Systems*, pages 1121–1131, 2018.

[60] Hongshun He, Deqiang Han, and Jean Dezert. Disagreement based semi-supervised learning approaches with belief functions. *Knowledge-Based Systems*, 193:105426, 2020.

[61] Saad Mohamad, Moamar Sayed-Mouchaweh, and Abdelhamid Bouchachia. Online active learning for human activity recognition from sensory data streams. *Neurocomputing*, 390:341–358, 2020.

[62] Fadi Dornaika. Active two phase collaborative representation classifier. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 13(4):1–10, 2019.

[63] Lili Yin, Huangang Wang, Wenhui Fan, Li Kou, Tingyu Lin, and Yingying Xiao. Incorporate active learning to semi-supervised industrial fault classification. *Journal of Process Control*, 78:88–97, 2019.

[64] Yazhou Yang and Marco Loog. A variance maximization criterion for active learning. *Pattern Recognition*, 78:358–370, 2018.

[65] Zhuochun Wu and Liye Xiao. A structure with density-weighted active learning-based model selection strategy and meteorological analysis for wind speed vector deterministic and probabilistic forecasting. *Energy*, 183:1178–1194, 2019.

[66] Shubhomoy Das, Weng-Keen Wong, Thomas Dietterich, Alan Fern, and Andrew Emmott. Discovering anomalies by incorporating feedback from an expert. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 14(4):1–32, 2020.

[67] Dominik Mautz, Wei Ye, Claudia Plant, and Christian Böhm. Non-redundant subspace clusterings with nr-kmeans and nr-dipmeans. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 14(5):1–24, 2020.

[68] Kaushik Ghosh, Sathish Natarajan, and Rajagopalan Srinivasan. Hierarchically distributed fault detection and identification through dempster–shafer evidence fusion. *Industrial & engineering chemistry research*, 50(15):9249–9269, 2011.

[69] Elaine Fehrman, Awaz K Muhammad, Evgeny M Mirkes, Vincent Egan, and Alexander N Gorban. The five factor model of personality and evaluation of drug consumption risk. In *Data science*, pages 231–242. Springer, 2017.

[70] Bearing Data Center. Case western reserve university seeded fault test data. *Case Western Reserve University: Cleveland, OH, USA*, 2015.

[71] Deepesh Agarwal, Manika Kumari, Babji Srinivasan, Prabhavathy Rajappan, and Jaishankar Chinnachamy. Fault diagnosis and degradation analysis of pmdc motors using fea based models. In *2020 IEEE International Conference on Power Electronics, Smart Grid and Renewable Energy (PESGRE2020)*, pages 1–6. IEEE, 2020.

[72] Tivadar Danka and Peter Horvath. modal: A modular active learning framework for python. *arXiv preprint arXiv:1805.00979*, 2018.

[73] Kaushalya Madhawa and Tsuyoshi Murata. Active learning for node classification: an evaluation. *Entropy*, 22(10):1164, 2020.

[74] Li Gao, Hong Yang, Chuan Zhou, Jia Wu, Shirui Pan, and Yue Hu. Active discriminative network representation learning. In *IJCAI International Joint Conference on Artificial Intelligence*, 2018.

[75] Xiaojin Zhu, Zoubin Ghahramani, and John D Lafferty. Semi-supervised learning using gaussian fields and harmonic functions. In *Proceedings of the 20th International conference on Machine learning (ICML-03)*, pages 912–919, 2003.

[76] Dengyong Zhou, Olivier Bousquet, Thomas Lal, Jason Weston, and Bernhard Schölkopf. Learning with local and global consistency. *Advances in neural information processing systems*, 16, 2003.

[77] Quanquan Gu and Jiawei Han. Towards active learning on graphs: An error bound minimization approach. In *2012 IEEE 12th International Conference on Data Mining*, pages 882–887. IEEE, 2012.

[78] Mustafa Bilgic, Lilyana Mihalkova, and Lise Getoor. Active learning for networked data. In *ICML*, 2010.

[79] Ming Ji and Jiawei Han. A variance minimization criterion to active learning on graphs. In *Artificial Intelligence and Statistics*, pages 556–564. PMLR, 2012.

[80] Hongyun Cai, Vincent W Zheng, and Kevin Chen-Chuan Chang. Active learning for graph embedding. *arXiv preprint arXiv:1705.05085*, 2017.

[81] Deepesh Agarwal, Pravesh Srivastava, Sergio Martin-del Campo, Balasubramaniam Natarajan, and Babji Srinivasan. Addressing practical challenges in active learning via a hybrid query strategy. *arXiv preprint arXiv:2110.03785*, 2021.

[82] Prithviraj Sen, Galileo Namata, Mustafa Bilgic, Lise Getoor, Brian Galligher, and Tina Eliassi-Rad. Collective classification in network data. *AI magazine*, 29(3):93–93, 2008.

[83] C Lee Giles, Kurt D Bollacker, and Steve Lawrence. Citeseer: An automatic citation indexing system. In *Proceedings of the third ACM conference on Digital libraries*, pages 89–98, 1998.

[84] Aric Hagberg, Pieter Swart, and Daniel S Chult. Exploring network structure, dynamics, and function using networkx. Technical report, Los Alamos National Lab.(LANL), Los Alamos, NM (United States), 2008.

[85] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32, 2019.

[86] Minjie Wang, Da Zheng, Zihao Ye, Quan Gan, Mufei Li, Xiang Song, Jinjing Zhou, Chao Ma, Lingfan Yu, Yu Gai, et al. Deep graph library: A graph-centric, highly-performant package for graph neural networks. *arXiv preprint arXiv:1909.01315*, 2019.

[87] Yildiray Yildiz. Cyberphysical human systems: An introduction to the special issue. *IEEE Control Systems Magazine*, 40(6):26–28, 2020.

[88] Daniel Kahneman. Maps of bounded rationality: Psychology for behavioral economics. *American economic review*, 93(5):1449–1475, 2003.

[89] Dan Ariely and Simon Jones. *Predictably irrational*. Harper Audio New York, NY, 2008.

[90] Syed Aliya Zahera and Rohit Bansal. Do investors exhibit behavioral biases in investment decision making? a systematic review. *Qualitative Research in Financial Markets*, 2018.

[91] Lindsay P Busby, Jesse L Courtier, and Christine M Glastonbury. Bias in radiology: the how and why of misses and misinterpretations. *Radiographics*, 38(1):236–247, 2018.

[92] Eoin D O'Sullivan and Susie Schofield. Cognitive bias in clinical medicine. *Journal of the Royal College of Physicians of Edinburgh*, 48(3):225–231, 2018.

[93] Emily Wall, Leslie M Blaha, Lyndsey Franklin, and Alex Endert. Warning, bias may occur: A proposed approach to detecting cognitive bias in interactive visual analytics. In *2017 IEEE Conference on Visual Analytics Science and Technology (VAST)*, pages 104–115. IEEE, 2017.

[94] Swati Mehta and Jaydip Chaudhari. The existence of behavioural factors among individual investors for investment decision in stock market: Evidence from indian stock market. *Global Journal of Research in Management*, 6(1):57, 2016.

[95] Fausto Giunchiglia, Mattia Zeni, and Enrico Big. Personal context recognition via reliable human-machine collaboration. In *2018 IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops)*, pages 379–384. IEEE, 2018.

[96] Alexander Amini, Ava P Soleimany, Wilko Schwarting, Sangeeta N Bhatia, and Daniela Rus. Uncovering and mitigating algorithmic bias through learned latent structure. In *Proceedings of the 2019 AAAI/ACM Conference on AI, Ethics, and Society*, pages 289–295, 2019.

[97] Vasileios Iosifidis and Eirini Ntoutsi. Dealing with bias via data augmentation in supervised learning scenarios. *Jo Bates Paul D. Clough Robert Jäschke*, 24, 2018.

[98] Mkhuseli Ngxande, Jules-Raymond Tapamo, and Michael Burke. Bias remediation in driver drowsiness detection systems using generative adversarial networks. *IEEE Access*, 8:55592–55601, 2020.

[99] Flavio P Calmon, Dennis Wei, Bhanukiran Vinzamuri, Karthikeyan Natesan Ramamurthy, and Kush R Varshney. Optimized pre-processing for discrimination prevention. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, pages 3995–4004, 2017.

[100] Mohsan Alvi, Andrew Zisserman, and Christoffer Nellåker. Turning a blind eye: Explicit removal of biases and variation from deep neural network embeddings. In *Proceedings of the European Conference on Computer Vision (ECCV) Workshops*, pages 0–0, 2018.

[101] Zeyu Wang, Klint Qinami, Ioannis Christos Karakozis, Kyle Genova, Prem Nair, Kenji Hata, and Olga Russakovsky. Towards fairness in visual recognition: Effective strate-

gies for bias mitigation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8919–8928, 2020.

[102] Brian Hu Zhang, Blake Lemoine, and Margaret Mitchell. Mitigating unwanted biases with adversarial learning. In *Proceedings of the 2018 AAAI/ACM Conference on AI, Ethics, and Society*, pages 335–340, 2018.

[103] Madumali Kalubowilage, Obdulia Covarrubias-Zambrano, Aruni P Malalasekera, Sebastian O Wendel, Hongwang Wang, Asanka S Yapa, Lauren Chlebanowski, Yubisela Toledo, Raquel Ortega, Katharine E Janik, et al. Early detection of pancreatic cancers in liquid biopsies by ultrasensitive fluorescence nanobiosensors. *Nanomedicine: Nanotechnology, Biology and Medicine*, 14(6):1823–1832, 2018.

[104] Jennifer S Blumenthal-Barby and Heather Krieger. Cognitive biases and heuristics in medical decision making: a critical review using a systematic search strategy. *Medical Decision Making*, 35(4):539–557, 2015.

[105] Satish K Mittal. Behavior biases and investment decision: theoretical and research framework. *Qualitative Research in Financial Markets*, 14(2):213–228, 2022.

[106] Jinesh Jain, Nidhi Walia, and Sanjay Gupta. Evaluation of behavioral biases affecting investment decision making of individual equity investors by fuzzy analytic hierarchy process. *Review of Behavioral Finance*, 12(3):297–314, 2020.

[107] Maqsood Ahmad, Syed Zulfiqar Ali Shah, and Yasar Abbass. The role of heuristic-driven biases in entrepreneurial strategic decision-making: evidence from an emerging economy. *Management Decision*, 59(3):669–691, 2021.

[108] Min Hun Lee, Daniel P Siewiorek, Asim Smailagic, Alexandre Bernardino, and Sergi Bermúdez i Badia. A human-ai collaborative approach for clinical decision making on rehabilitation assessment. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*, pages 1–14, 2021.

[109] Wansoo Kim, Luka Peternel, Marta Lorenzini, Jan Babič, and Arash Ajoudani. A human-robot collaboration framework for improving ergonomics during dexterous operation of power tools. *Robotics and Computer-Integrated Manufacturing*, 68:102084, 2021.

[110] Quan Liu, Zhihao Liu, Bo Xiong, Wenjun Xu, and Yang Liu. Deep reinforcement learning-based safe interaction for industrial human-robot collaboration using intrinsic reward function. *Advanced Engineering Informatics*, 49:101360, 2021.

[111] Konrad Sowa, Aleksandra Przegalinska, and Leon Ciechanowski. Cobots in knowledge work: Human–ai collaboration in managerial professions. *Journal of Business Research*, 125:135–142, 2021.

[112] Daniel N Cassenti and Lance M Kaplan. Robust uncertainty representation in human-ai collaboration. In *Artificial Intelligence and Machine Learning for Multi-Domain Operations Applications III*, volume 11746, pages 249–262. SPIE, 2021.

[113] Samuel Budd, Emma C Robinson, and Bernhard Kainz. A survey on active learning and human-in-the-loop deep learning for medical image analysis. *Medical Image Analysis*, 71:102062, 2021.

[114] Alex Wang. Interplay of investors' financial knowledge and risk taking. *The journal of behavioral finance*, 10(4):204–213, 2009.

[115] Deepesh Agarwal, Obdulia Covarrubias-Zambrano, Stefan H Bossmann, and Balasubramaniam Natarajan. Early Detection of Pancreatic Cancers Using Liquid Biopsies and Hierarchical Decision Structure. *IEEE Journal of Translational Engineering in Health and Medicine*, 10:1–8, 2022.

[116] George Em Karniadakis, Ioannis G Kevrekidis, Lu Lu, Paris Perdikaris, Sifan Wang, and Liu Yang. Physics-informed machine learning. *Nature Reviews Physics*, 3(6): 422–440, 2021.

[117] Kamesh Krishnamurthy and Daniel T Laskowitz. Cellular and molecular mechanisms of secondary neuronal injury following traumatic brain injury. 2015.

[118] Helen M Bramlett and W Dalton Dietrich. Long-term consequences of traumatic brain injury: current status of potential mechanisms of injury and neurological outcomes. *Journal of neurotrauma*, 32(23):1834–1848, 2015.

[119] Dragan Pavlovic, Sandra Pekic, Marko Stojanovic, and Vera Popovic. Traumatic brain injury: neuropathological, neurocognitive and neurobehavioral sequelae. *Pituitary*, 22: 270–282, 2019.

[120] Rodger Ll Wood and Andrew Worthington. Neurobehavioral abnormalities associated with executive dysfunction after traumatic brain injury. *Frontiers in behavioral neuroscience*, 11:195, 2017.

[121] Andrew IR Maas, David K Menon, Geoffrey T Manley, Mathew Abrams, Cecilia Åkerlund, Nada Andelic, Marcel Aries, Tom Bashford, Michael J Bell, Yelena G Bodien, et al. Traumatic brain injury: progress and challenges in prevention, clinical care, and research. *The Lancet Neurology*, 21(11):1004–1060, 2022.

[122] Damian Szklarczyk, Annika L Gable, David Lyon, Alexander Junge, Stefan Wyder, Jaime Huerta-Cepas, Milan Simonovic, Nadezhda T Doncheva, John H Morris, Peer Bork, et al. String v11: protein–protein association networks with increased coverage, supporting functional discovery in genome-wide experimental datasets. *Nucleic acids research*, 47(D1):D607–D613, 2019.

[123] Deepesh Agarwal, Sangeetha Ramu, Murali Nagarajan, Prabhavathy Rajappan, Jaishankar Chinnachamy, and Babji Srinivasan. Ipms for load sharing, monitoring and diagnosis of electrical loads in afvs. In *2020 7th International Conference on Signal Processing and Integrated Networks (SPIN)*, pages 956–961. IEEE, 2020.

[124] Sheng Shen, Hao Lu, Mohammadkazem Sadoughi, Chao Hu, Venkat Nemani, Adam Thelen, Keith Webster, Matthew Darr, Jeff Sidon, and Shawn Kenny. A physics-

informed deep learning approach for bearing fault detection. *Engineering Applications of Artificial Intelligence*, 103:104295, 2021.

[125] Sriram Anbalagan, Balasubramaniam Natarajan, and Babji Srinivasan. A physics based deep learning approach for cross domain bearing fault detection. In *2023 IEEE Kansas Power and Energy Conference (KPEC)*, pages 1–4. IEEE, 2023.

[126] Konstantinos C Gryllias and Ioannis A Antoniadis. A support vector machine approach based on physical model training for rolling element bearing fault detection in industrial environments. *Engineering Applications of Artificial Intelligence*, 25(2):326–344, 2012.

[127] Sayanjit Singha Roy, Sayantan Dey, and Soumya Chatterjee. Autocorrelation aided random forest classifier-based bearing fault detection framework. *IEEE Sensors Journal*, 20(18):10792–10800, 2020.

[128] Isaac Triguero, Diego García-Gil, Jesús Maillo, Julián Luengo, Salvador García, and Francisco Herrera. Transforming big data into smart data: An insight on the use of the k-nearest neighbors algorithm to obtain quality data. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 9(2):e1289, 2019.

[129] Bo Zhao, Xianmin Zhang, Hai Li, and Zhuobo Yang. Intelligent fault diagnosis of rolling bearings based on normalized cnn considering data imbalance and variable working conditions. *Knowledge-Based Systems*, 199:105971, 2020.

[130] Han Liu, Jianzhong Zhou, Yang Zheng, Wei Jiang, and Yuncheng Zhang. Fault diagnosis of rolling bearings with recurrent neural network-based autoencoders. *ISA transactions*, 77:167–178, 2018.

[131] Rui Yang, Mengjie Huang, Qidong Lu, and Maiying Zhong. Rotating machinery fault diagnosis using long-short-term memory recurrent neural network. *IFAC-PapersOnLine*, 51(24):228–232, 2018.

[132] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N

Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[133] Xianjun Du, Liangliang Jia, and Izaz Ul Haq. Fault diagnosis based on spbo-sdae and transformer neural network for rotating machinery. *Measurement*, 188:110545, 2022.

[134] Zihang Dai, Zhilin Yang, Yiming Yang, Jaime Carbonell, Quoc V Le, and Ruslan Salakhutdinov. Transformer-xl: Attentive language models beyond a fixed-length context. *arXiv preprint arXiv:1901.02860*, 2019.

[135] Katharina Rombach, Gabriel Michau, and Olga Fink. Contrastive learning for fault detection and diagnostics in the context of changing operating conditions and novel fault types. *Sensors*, 21(10):3550, 2021.

[136] Roozbeh Razavi-Far, Ehsan Hallaji, Maryam Farajzadeh-Zanjani, and Mehrdad Saif. A semi-supervised diagnostic framework based on the surface estimation of faulty distributions. *IEEE Transactions on Industrial Informatics*, 15(3):1277–1286, 2018.

[137] Huan Wang, Zhiliang Liu, Yipei Ge, and Dandan Peng. Self-supervised signal representation learning for machinery fault diagnosis under limited annotation data. *Knowledge-Based Systems*, 239:107978, 2022.

[138] Florian Schroff, Dmitry Kalenichenko, and James Philbin. Facenet: A unified embedding for face recognition and clustering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 815–823, 2015.

[139] Debidatta Dwibedi, Yusuf Aytar, Jonathan Tompson, Pierre Sermanet, and Andrew Zisserman. With a little help from my friends: Nearest-neighbor contrastive learning of visual representations. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9588–9597, 2021.

[140] Lili Yin, Huangang Wang, Wenhui Fan, Li Kou, Tingyu Lin, and Yingying Xiao. Incorporate active learning to semi-supervised industrial fault classification. *Journal of Process Control*, 78:88–97, 2019.

[141] Jiayu Chen, Chen Lu, and Hang Yuan. Bearing fault diagnosis based on active learning and random forest. *Vibroengineering PROCEDIA*, 5:321–326, 2015.

[142] Solomon Kullback and Richard A Leibler. On information and sufficiency. *The annals of mathematical statistics*, 22(1):79–86, 1951.

[143] Jacob Devlin Ming-Wei Chang Kenton and Lee Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of naacL-HLT*, volume 1, page 2, 2019.

[144] Wonho Jung, Seong-Hu Kim, Sung-Hyun Yun, Jaewoong Bae, and Yong-Hwa Park. Vibration, acoustic, temperature, and motor current dataset of rotating machine under varying operating conditions for fault diagnosis. *Data in Brief*, 48:109049, 2023.

[145] Mert Sehri, Patrick Dumond, and Michel Bouchard. University of ottawa constant load and speed rolling-element bearing vibration and acoustic fault signature datasets. *Data in Brief*, page 109327, 2023.

[146] Nguyen Duc Thuan and Hoang Si Hong. Hust bearing: a practical dataset for ball bearing fault diagnosis. *arXiv preprint arXiv:2302.12533*, 2023.

[147] Survival rates for pancreatic cancer. https://www.cancer.org/cancer/pancreatic-cancer/detection-diagnosis-staging/survival-rates.html, . Accessed: 2022-05-24.

[148] Rebecca L Siegel, Kimberly D Miller, and Ahmedin Jemal. Cancer statistics, 2020. *CA: a cancer journal for clinicians*, 66(1):7–30, 2020.

[149] C Güngör, BT Hofmann, G Wolters-Eisfeld, and M Bockhorn. Pancreatic cancer. *British journal of pharmacology*, 171(4):849–858, 2014.

[150] Shinichi Fukushige and Akira Horii. Road to early detection of pancreatic cancer: Attempts to utilize epigenetic biomarkers. *Cancer letters*, 342(2):231–237, 2014.

[151] Tamotsu Kuroki and Susumu Eguchi. No-touch isolation techniques for pancreatic cancer. *Surgery today*, 47(1):8–13, 2017.

[152] Stefan Boeck, Donna Pauler Ankerst, and Volker Heinemann. The role of adjuvant chemotherapy for patients with resected pancreatic cancer: systematic review of randomized controlled trials and meta-analysis. *Oncology*, 72(5-6):314–321, 2007.

[153] Pancreatic cancer. https://www.mayoclinic.org/diseases-conditions/pancreatic-cancer/diagnosis-treatment/drc-20355427, . Accessed: 2022-05-20.

[154] Stefan H Bossmann. Liquid biopsies for early cancer detection. In *Biomaterials for Cancer Therapeutics*, pages 233–259. Elsevier, 2020.

[155] Emily Crowley, Federica Di Nicolantonio, Fotios Loupakis, and Alberto Bardelli. Liquid biopsy: monitoring cancer-genetics in the blood. *Nature reviews Clinical oncology*, 10(8):472–484, 2013.

[156] Nabiollah Kamyabi, Vincent Bernard, and Anirban Maitra. Liquid biopsies in pancreatic cancer. *Expert review of anticancer therapy*, 19(10):869–878, 2019.

[157] Miles W Grunvald, Richard A Jacobson, Timothy M Kuzel, Sam G Pappas, and Ashiq Masood. Current status of circulating tumor dna liquid biopsy in pancreatic cancer. *International Journal of Molecular Sciences*, 21(20):7651, 2020.

[158] Tetsuhiro Okada, Yusuke Mizukami, Akihiro Hayashi, Hidemasa Kawabata, Hiroki Sato, Tooru Kawamoto, Takuma Goto, Kenzui Taniue, Yusuke Ono, Hidenori Karasaki, et al. Liquid biopsy of pancreatic tumors: Challenges for early detection and surveillance based on the molecular landscape during early carcinogenesis. *Suizo*, 35(4):302–312, 2020.

[159] Marta Toledano-Fonseca, M Teresa Cano, Elizabeth Inga, Rosa Rodríguez-Alonso, M Auxiliadora Gómez-España, Silvia Guil-Luna, Rafael Mena-Osuna, Juan R de la

Haba-Rodríguez, Antonio Rodríguez-Ariza, and Enrique Aranda. Circulating cell-free dna-based liquid biopsy markers for the non-invasive prognosis and monitoring of metastatic pancreatic cancer. *Cancers*, 12(7):1754, 2020.

[160] Victoria Heredia-Soto, Nuria Rodríguez-Salas, and Jaime Feliu. Liquid biopsy in pancreatic cancer: Are we ready to apply it in the clinical practice? *Cancers*, 13(8):1986, 2021.

[161] Jun Hou, XueTao Li, and Ke-Ping Xie. Coupled liquid biopsy and bioinformatics for pancreatic cancer early detection and precision prognostication. *Molecular Cancer*, 20 (1):1–12, 2021.

[162] YounJeong Choi and Christina Kendziorski. Statistical methods for gene set co-expression analysis. *Bioinformatics*, 25(21):2780–2786, 2009.

[163] Hongwang Wang, Dinusha N Udukala, Thilani N Samarakoon, Matthew T Basel, Mausam Kalita, Gayani Abayaweera, Harshi Manawadu, Aruni Malalasekera, Colette Robinson, David Villanueva, et al. Nanoplatforms for highly sensitive fluorescence detection of cancer-related proteases. *Photochemical & Photobiological Sciences*, 13 (2):231–240, 2014.

[164] Dinusha N Udukala, Hongwang Wang, Sebastian O Wendel, Aruni P Malalasekera, Thilani N Samarakoon, Asanka S Yapa, Gayani Abayaweera, Matthew T Basel, Pamela Maynez, Raquel Ortega, et al. Early breast cancer screening using iron/iron oxide-based nanoplatforms with sub-femtomolar limits of detection. *Beilstein journal of nanotechnology*, 7(1):364–373, 2016.

[165] Aruni P Malalasekera, Hongwang Wang, Thilani N Samarakoon, Dinusha N Udukala, Asanka S Yapa, Raquel Ortega, Tej B Shrestha, Hamad Alshetaiwi, Emily J McLaurin, Deryl L Troyer, et al. A nanobiosensor for the detection of arginase activity. *Nanomedicine: Nanotechnology, Biology and Medicine*, 13(2):383–390, 2017.

[166] Dinusha N Udukala, Sebastian O Wendel, Hongwang Wang, Asanka S Yapa, Obdulia Covarrubias-Zambrano, Katharine Janik, Gary Gadbury, Deryl L Troyer, and Stefan H Bossmann. Early detection of non-small cell lung cancer in liquid biopsies by ultra-sensitive protease activity analysis. *Journal of Cancer Metastasis and Treatment*, 6, 2020.

[167] Bernard L Welch. The generalization of 'student's'problem when several different population varlances are involved. *Biometrika*, 34(1-2):28–35, 1947.

[168] Christopher S Coffey and Stacey S Cofield. Parametric linear models. In *DNA Microarrays and Related Genomics Techniques*, pages 241–262. Chapman and Hall/CRC, 2005.

[169] Roger Alan Stein, Patricia A Jaques, and Joao Francisco Valiati. An analysis of hierarchical text classification using word embeddings. *Information Sciences*, 471:216–232, 2019.

[170] Ahmed Ahmim, Leandros Maglaras, Mohamed Amine Ferrag, Makhlouf Derdour, and Helge Janicke. A novel hierarchical intrusion detection system based on decision tree and rules-based models. In *2019 15th International Conference on Distributed Computing in Sensor Systems (DCOSS)*, pages 228–233. IEEE, 2019.

[171] Mukund Subramaniyan, Anders Skoogh, Azam Sheikh Muhammad, Jon Bokrantz, Björn Johansson, and Christoph Roser. A generic hierarchical clustering approach for detecting bottlenecks in manufacturing. *Journal of Manufacturing Systems*, 55: 143–158, 2020.

[172] Paweł Pławiak, Moloud Abdar, Joanna Pławiak, Vladimir Makarenkov, and U Rajendra Acharya. Dghnl: A new deep genetic hierarchical network of learners for prediction of credit scoring. *Information Sciences*, 516:401–418, 2020.

[173] Wengong Jin, Regina Barzilay, and Tommi Jaakkola. Hierarchical generation of molec-

ular graphs using structural motifs. In *International Conference on Machine Learning*, pages 4839–4848. PMLR, 2020.

[174] Shiwen Shen, Simon X Han, Denise R Aberle, Alex A Bui, and William Hsu. An interpretable deep hierarchical semantic convolutional neural network for lung nodule malignancy classification. *Expert systems with applications*, 128:84–95, 2019.

[175] Rodolfo M Pereira, Diego Bertolini, Lucas O Teixeira, Carlos N Silla Jr, and Yandre MG Costa. Covid-19 identification in chest x-ray images on flat and hierarchical classification scenarios. *Computer methods and programs in biomedicine*, 194:105532, 2020.

[176] Farhat Afza, Muhammad Sharif, Mamta Mittal, Muhammad Attique Khan, and D Jude Hemanth. A hierarchical three-step superpixels and deep learning framework for skin lesion classification. *Methods*, 202:88–102, 2022.

[177] Jun Pyo Kim, Jeonghun Kim, Yu Hyun Park, Seong Beom Park, Jin San Lee, Sole Yoo, Eun-Joo Kim, Hee Jin Kim, Duk L Na, Jesse A Brown, et al. Machine learning based hierarchical classification of frontotemporal dementia and alzheimer's disease. *NeuroImage: Clinical*, 23:101811, 2019.

[178] Dimitrios Schizas, Nikolaos Charalampakis, Christo Kole, Panagiota Economopoulou, Evangelos Koustas, Efthymios Gkotsis, Dimitrios Ziogas, Amanda Psyrri, and Michalis V Karamouzis. Immunotherapy for pancreatic cancer: A 2020 update. *Cancer treatment reviews*, 86:102016, 2020.

[179] Biospecimen. http://www.kucancercenter.org/patient-care/biospecimen-bank/staff. Accessed: 2022-05-26.

[180] Dennis D Boos and Leonard A Stefanski. P-value precision and reproducibility. *The American Statistician*, 65(4):213–221, 2011.

[181] Avjinder S Kaler and Larry C Purcell. Estimation of a significance threshold for genome-wide association studies. *BMC genomics*, 20(1):1–8, 2019.

[182] Christoph Lippert, Jennifer Listgarten, Ying Liu, Carl M Kadie, Robert I Davidson, and David Heckerman. Fast linear mixed models for genome-wide association studies. *Nature methods*, 8(10):833–835, 2011.

[183] Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al. Scikit-learn: Machine learning in python. *the Journal of machine Learning research*, 12:2825–2830, 2011.

[184] Jayasree Chakraborty, Liana Langdon-Embry, Kristen M Cunanan, Joanna G Escalon, Peter J Allen, Maeve A Lowery, Eileen M O'Reilly, Mithat Gönen, Richard G Do, and Amber L Simpson. Preliminary study of tumor heterogeneity in imaging predicts two year survival in pancreatic cancer patients. *PloS one*, 12(12):e0188022, 2017.

[185] Guangtao Ge and G William Wong. Classification of premalignant pancreatic cancer mass-spectrometry data using decision tree ensembles. *BMC bioinformatics*, 9(1):1–12, 2008.

[186] Wismaji Sadewo, Zuherman Rustam, Hamidah Hamidah, and Alifah Roudhoh Chusmarsyah. Pancreatic cancer early detection using twin support vector machine based on kernel. *Symmetry*, 12(4):667, 2020.

[187] Chunyang Li, Xiaoxi Zeng, Haopeng Yu, Yonghong Gu, and Wei Zhang. Identification of hub genes with diagnostic values in pancreatic cancer by bioinformatics analyses and supervised learning methods. *World journal of surgical oncology*, 16(1):1–12, 2018.

[188] Sudeepa Bhattacharyya, Eric R Siegel, Gloria M Petersen, Suresh T Chari, Larry J Suva, and Randy S Haun. Diagnosis of pancreatic cancer using serum proteomic profiling. *Neoplasia*, 6(5):674–686, 2004.

[189] Ie-Ming Shih, Yeh Wang, and Tian-Li Wang. The origin of ovarian cancer species and precancerous landscape. *The American journal of pathology*, 191(1):26–39, 2021.

[190] Mara H Rendi, Rochelle L Garcia, and Don S Dizon. Epithelial carcinoma of the ovary, fallopian tube, and peritoneum: histopathology. *Available via www. uptodate. com. Published*, 2016.

[191] David D Bowtell, Steffen Böhm, Ahmed A Ahmed, Paul-Joseph Aspuria, Robert C Bast Jr, Valerie Beral, Jonathan S Berek, Michael J Birrer, Sarah Blagden, Michael A Bookman, et al. Rethinking ovarian cancer ii: reducing mortality from high-grade serous ovarian cancer. *Nature reviews Cancer*, 15(11):668–679, 2015.

[192] Rebecca L Siegel, Kimberly D Miller, Hannah E Fuchs, Ahmedin Jemal, et al. Cancer statistics, 2021. *Ca Cancer J Clin*, 71(1):7–33, 2021.

[193] Jonathan S Berek, Sean T Kehoe, Lalit Kumar, and Michael Friedlander. Cancer of the ovary, fallopian tube, and peritoneum. *International journal of gynecology & obstetrics*, 143:59–78, 2018.

[194] Barbara A Goff, Lynn S Mandel, Charles W Drescher, Nicole Urban, Shirley Gough, Kristi M Schurman, Joshua Patras, Barry S Mahony, and M Robyn Andersen. Development of an ovarian cancer symptom index: possibilities for earlier detection. *Cancer*, 109(2):221–227, 2007.

[195] Christine Stewart, Christine Ralyea, and Suzy Lockwood. Ovarian cancer: an integrated review. In *Seminars in oncology nursing*, volume 35, pages 151–156. Elsevier, 2019.

[196] Lindsey A Torre, Britton Trabert, Carol E DeSantis, Kimberly D Miller, Goli Samimi, Carolyn D Runowicz, Mia M Gaudet, Ahmedin Jemal, and Rebecca L Siegel. Ovarian cancer statistics, 2018. *CA: a cancer journal for clinicians*, 68(4):284–296, 2018.

[197] Chyke A Doubeni, Douglas A Corley, Virginia P Quinn, Christopher D Jensen, Ann G Zauber, Michael Goodman, Jill R Johnson, Shivan J Mehta, Tracy A Becerra, Wei K Zhao, et al. Effectiveness of screening colonoscopy in reducing the risk of death from right and left colon cancer: a large community-based study. *Gut*, 67(2):291–298, 2018.

[198] Kristen A Matteson, Camille Gunderson, and Debra L Richardson. The role of the obstetrician-gynecologist in the early detection of epithelial ovarian cancer in women at average risk. *OBSTETRICS AND GYNECOLOGY*, 130(3):E146–E149, 2017.

[199] David C Grossman, Susan J Curry, Douglas K Owens, Michael J Barry, Karina W Davidson, Chyke A Doubeni, John W Epling, Alex R Kemper, Alex H Krist, Ann E Kurth, et al. Screening for ovarian cancer: Us preventive services task force recommendation statement. *Jama*, 319(6):588–594, 2018.

[200] Lei Chang, Jie Ni, Ying Zhu, Bairen Pang, Peter Graham, Hao Zhang, and Yong Li. Liquid biopsy in ovarian cancer: Recent advances in circulating extracellular vesicle detection for early diagnosis and monitoring progression. *Theranostics*, 9(14):4130, 2019.

[201] J Barrett, V Jenkins, V Farewell, U Menon, I Jacobs, J Kilkerr, A Ryan, C Langridge, L Fallowfield, and UKCTOCS trialists. Psychological morbidity associated with ovarian cancer screening: results from more than 23 000 women in the randomised trial of ovarian cancer screening (ukctocs). *BJOG: An International Journal of Obstetrics & Gynaecology*, 121(9):1071–1079, 2014.

[202] Alejandro Barredo Arrieta, Natalia Díaz-Rodríguez, Javier Del Ser, Adrien Bennetot, Siham Tabik, Alberto Barbado, Salvador García, Sergio Gil-López, Daniel Molina, Richard Benjamins, et al. Explainable artificial intelligence (xai): Concepts, taxonomies, opportunities and challenges toward responsible ai. *Information fusion*, 58: 82–115, 2020.

[203] Scott M Lundberg and Su-In Lee. A unified approach to interpreting model predictions. *Advances in neural information processing systems*, 30, 2017.

[204] Jakob Gawlikowski, Cedrique Rovile Njieutcheu Tassi, Mohsin Ali, Jongseok Lee, Matthias Humt, Jianxiang Feng, Anna Kruspe, Rudolph Triebel, Peter Jung, Ribana

Roscher, et al. A survey of uncertainty in deep neural networks. *Artificial Intelligence Review*, pages 1–77, 2023.

[205] Pengzhen Ren, Yun Xiao, Xiaojun Chang, Po-Yao Huang, Zhihui Li, Brij B Gupta, Xiaojiang Chen, and Xin Wang. A survey of deep active learning. *ACM computing surveys (CSUR)*, 54(9):1–40, 2021.

[206] Deepesh Agarwal and Balasubramaniam Natarajan. Active learning for node classification using a convex optimization approach. In *2022 IEEE Eighth International Conference on Big Data Computing Service and Applications (BigDataService)*, pages 96–102. IEEE, 2022.

[207] Wenchong He and Zhe Jiang. A survey on uncertainty quantification methods for deep neural networks: An uncertainty source perspective. *arXiv preprint arXiv:2302.13425*, 2023.

[208] Jay Nandy, Wynne Hsu, and Mong Li Lee. Towards maximizing the representation gap between in-domain & out-of-distribution examples. *Advances in Neural Information Processing Systems*, 33:9239–9250, 2020.

[209] Theodoros Tsiligkaridis. Information robust dirichlet networks for predictive uncertainty estimation, April 8 2021. US Patent App. 17/064,046.

[210] Vishal Thanvantri Vasudevan, Abhinav Sethy, and Alireza Roshan Ghias. Towards better confidence estimation for neural models. In *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 7335–7339. IEEE, 2019.

[211] Takumi Kawashima, Qina Yu, Akari Asai, Daiki Ikami, and Kiyoharu Aizawa. The aleatoric uncertainty estimation using a separate formulation with virtual residuals. In *2020 25th International Conference on Pattern Recognition (ICPR)*, pages 1438–1445. IEEE, 2021.

[212] Tiago Ramalho and Miguel Miranda. Density estimation in representation space to predict model uncertainty. In *Engineering Dependable and Secure Machine Learning Systems: Third International Workshop, EDSMLS 2020, New York City, NY, USA, February 7, 2020, Revised Selected Papers 3*, pages 84–96. Springer, 2020.

[213] Yen-Chang Hsu, Yilin Shen, Hongxia Jin, and Zsolt Kira. Generalized odin: Detecting out-of-distribution image without learning from out-of-distribution data. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10951–10960, 2020.

[214] Jinsol Lee and Ghassan AlRegib. Gradients as a measure of uncertainty in neural networks. In *2020 IEEE International Conference on Image Processing (ICIP)*, pages 2416–2420. IEEE, 2020.

[215] Antonio Loquercio, Mattia Segu, and Davide Scaramuzza. A general framework for uncertainty estimation in deep learning. *IEEE Robotics and Automation Letters*, 5(2): 3153–3160, 2020.

[216] Aryan Mobiny, Pengyu Yuan, Supratik K Moulik, Naveen Garg, Carol C Wu, and Hien Van Nguyen. Dropconnect is effective in modeling uncertainty of bayesian deep networks. *Scientific reports*, 11(1):5458, 2021.

[217] Christopher Nemeth and Paul Fearnhead. Stochastic gradient markov chain monte carlo. *Journal of the American Statistical Association*, 116(533):433–450, 2021.

[218] Florian Wenzel, Kevin Roth, Bastiaan S Veeling, Jakub Swiatkowski, Linh Tran, Stephan Mandt, Jasper Snoek, Tim Salimans, Rodolphe Jenatton, and Sebastian Nowozin. How good is the bayes posterior in deep neural networks really? *arXiv preprint arXiv:2002.02405*, 2020.

[219] Alexander Immer, Maciej Korzepa, and Matthias Bauer. Improving predictions of bayesian neural nets via local linearization. In *International conference on artificial intelligence and statistics*, pages 703–711. PMLR, 2021.

[220] Marius Hobbhahn, Agustinus Kristiadi, and Philipp Hennig. Fast predictive uncertainty for classification with bayesian deep networks. In *Uncertainty in Artificial Intelligence*, pages 822–832. PMLR, 2022.

[221] Omer Sagi and Lior Rokach. Ensemble learning: A survey. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 8(4):e1249, 2018.

[222] Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. Simple and scalable predictive uncertainty estimation using deep ensembles. *Advances in neural information processing systems*, 30, 2017.

[223] Florian Wenzel, Jasper Snoek, Dustin Tran, and Rodolphe Jenatton. Hyperparameter ensembles for robustness and uncertainty quantification. *Advances in Neural Information Processing Systems*, 33:6514–6527, 2020.

[224] Yeming Wen, Dustin Tran, and Jimmy Ba. Batchensemble: an alternative approach to efficient ensemble and lifelong learning. *arXiv preprint arXiv:2002.06715*, 2020.

[225] Marton Havasi, Rodolphe Jenatton, Stanislav Fort, Jeremiah Zhe Liu, Jasper Snoek, Balaji Lakshminarayanan, Andrew M. Dai, and Dustin Tran. Training independent subnetworks for robust prediction. In *International Conference on Learning Representations*, 2021.

[226] Murat Seckin Ayhan and Philipp Berens. Test-time data augmentation for estimation of heteroscedastic aleatoric uncertainty in deep neural networks. In *Medical Imaging with Deep Learning*, 2022.

[227] Divya Shanmugam, Davis Blalock, Guha Balakrishnan, and John Guttag. Better aggregation in test-time augmentation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 1214–1223, 2021.

[228] Alexander Lyzhov, Yuliya Molchanova, Arsenii Ashukha, Dmitry Molchanov, and Dmitry Vetrov. Greedy policy search: A simple baseline for learnable test-time aug-

mentation. In *Conference on Uncertainty in Artificial Intelligence*, pages 1308–1317. PMLR, 2020.

[229] QHwan Kim, Joon-Hyuk Ko, Sunghoon Kim, Nojun Park, and Wonho Jhe. Bayesian neural network with pretrained protein embedding enhances prediction accuracy of drug-protein interaction. *Bioinformatics*, 37(20):3428–3435, 2021.

[230] Alice Hein, Stefan Röhrl, Thea Grobel, Manuel Lengl, Nawal Hafez, Martin Knopp, Christian Klenk, Dominik Heim, Oliver Hayden, and Klaus Diepold. A comparison of uncertainty quantification methods for active learning in image classification. In *2022 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8. IEEE, 2022.

[231] Muhammad Ahtazaz Ahsan, Adnan Qayyum, Adeel Razi, and Junaid Qadir. An active learning method for diabetic retinopathy classification with uncertainty quantification. *Medical & Biological Engineering & Computing*, 60(10):2797–2811, 2022.

[232] Yue Cao and Yang Shen. Bayesian active learning for optimization and uncertainty quantification in protein docking. *Journal of chemical theory and computation*, 16(8): 5334–5347, 2020.

[233] Osman Mamun, MFN Taufique, Madison Wenzlick, Jeffrey Hawk, and Ram Devanathan. Uncertainty quantification for bayesian active learning in rupture life prediction of ferritic steels. *Scientific Reports*, 12(1):2083, 2022.

[234] Jochen Gast and Stefan Roth. Lightweight probabilistic deep networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3369–3378, 2018.

[235] Xavier Boyen and Daphne Koller. Tractable inference for complex stochastic processes. *arXiv preprint arXiv:1301.7362*, 2013.

[236] Roy De Maesschalck, Delphine Jouan-Rimbaud, and Désiré L Massart. The mahalanobis distance. *Chemometrics and intelligent laboratory systems*, 50(1):1–18, 2000.

[237] Anselm Blumer, Andrzej Ehrenfeucht, David Haussler, and Manfred K Warmuth. Learnability and the vapnik-chervonenkis dimension. *Journal of the ACM (JACM)*, 36(4):929–965, 1989.

[238] Steve Hanneke and Liu Yang. Surrogate losses in passive and active learning. 2019.

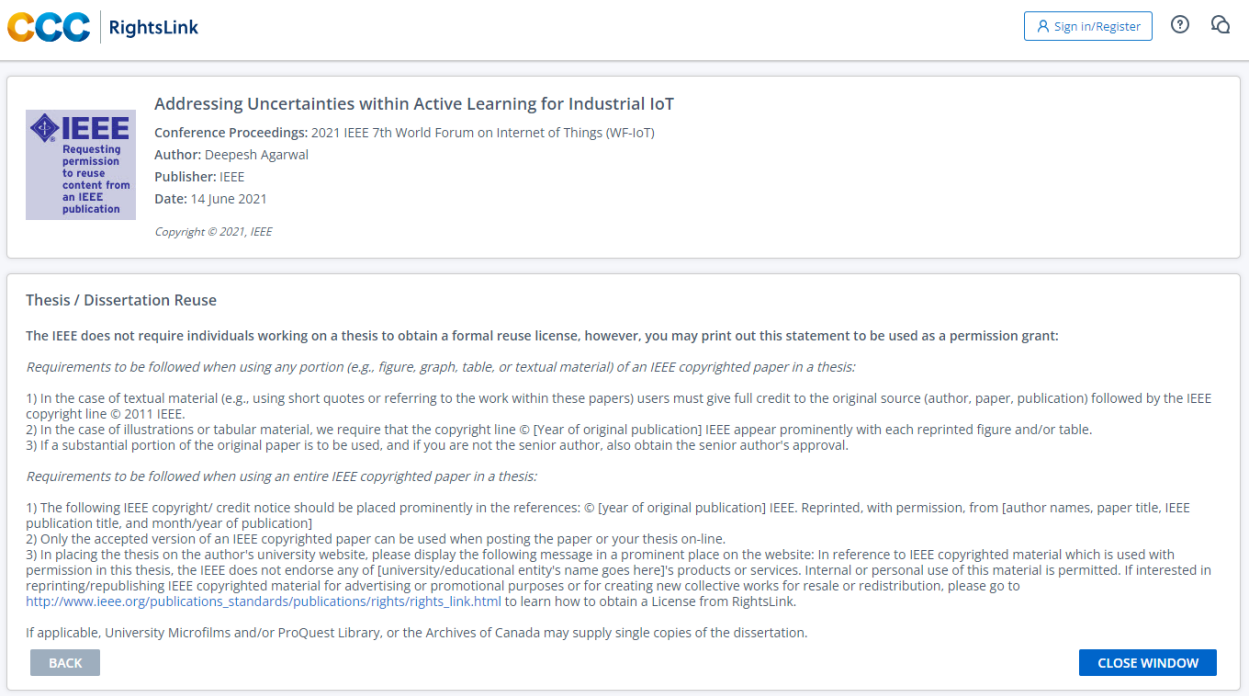
[239] Brett Lantz. *Machine learning with R: expert techniques for predictive modeling*. Packt publishing ltd, 2019.

[240] Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q Weinberger. On calibration of modern neural networks. In *International conference on machine learning*, pages 1321–1330. PMLR, 2017.

[241] Matthias Minderer, Josip Djolonga, Rob Romijnders, Frances Hubis, Xiaohua Zhai, Neil Houlsby, Dustin Tran, and Mario Lucic. Revisiting the calibration of modern neural networks. *Advances in Neural Information Processing Systems*, 34:15682–15694, 2021.

Appendix A

Reuse permissions from publishers



CCC RightsLink

Sign in/Register

IEEE Requesting permission to reuse content from an IEEE publication

Addressing Uncertainties within Active Learning for Industrial IoT

Conference Proceedings: 2021 IEEE 7th World Forum on Internet of Things (WF-IoT)

Author: Deepesh Agarwal

Publisher: IEEE

Date: 14 June 2021

Copyright © 2021, IEEE

Thesis / Dissertation Reuse

The IEEE does not require individuals working on a thesis to obtain a formal reuse license, however, you may print out this statement to be used as a permission grant:

Requirements to be followed when using any portion (e.g., figure, graph, table, or textual material) of an IEEE copyrighted paper in a thesis:

1) In the case of textual material (e.g., using short quotes or referring to the work within these papers) users must give full credit to the original source (author, paper, publication) followed by the IEEE copyright line © 2011 IEEE.
2) In the case of illustrations or tabular material, we require that the copyright line © [Year of original publication] IEEE appear prominently with each reprinted figure and/or table.
3) If a substantial portion of the original paper is to be used, and if you are not the senior author, also obtain the senior author's approval.

Requirements to be followed when using an entire IEEE copyrighted paper in a thesis:

1) The following IEEE copyright/ credit notice should be placed prominently in the references: © [year of original publication] IEEE. Reprinted, with permission, from [author names, paper title, IEEE publication title, and month/year of publication]
2) Only the accepted version of an IEEE copyrighted paper can be used when posting the paper or your thesis on-line.
3) In placing the thesis on the author's university website, please display the following message in a prominent place on the website: In reference to IEEE copyrighted material which is used with permission in this thesis, the IEEE does not endorse any of [university/educational entity's name goes here]'s products or services. Internal or personal use of this material is permitted. If interested in reprinting/republishing IEEE copyrighted material for advertising or promotional purposes or for creating new collective works for resale or redistribution, please go to http://www.ieee.org/publications_standards/publications/rights/rights_link.html to learn how to obtain a License from RightsLink.

If applicable, University Microfilms and/or ProQuest Library, or the Archives of Canada may supply single copies of the dissertation.

BACK

CLOSE WINDOW

CCC RightsLink

Sign in/Register

IEEE Requesting permission to reuse content from an IEEE publication

Impacts of Behavioral Biases on Active Learning Strategies

Conference Proceedings: 2022 International Conference on Artificial Intelligence in Information and Communication (ICAIIC)
Author: Deepesh Agarwal
Publisher: IEEE
Date: 21 February 2022

Thesis / Dissertation Reuse

The IEEE does not require individuals working on a thesis to obtain a formal reuse license, however, you may print out this statement to be used as a permission grant:

Requirements to be followed when using any portion (e.g., figure, graph, table, or textual material) of an IEEE copyrighted paper in a thesis:

1) In the case of textual material (e.g., using short quotes or referring to the work within these papers) users must give full credit to the original source (author, paper, publication) followed by the IEEE copyright line © 2011 IEEE.
2) In the case of illustrations or tabular material, we require that the copyright line © [Year of original publication] IEEE appear prominently with each reprinted figure and/or table.
3) If a substantial portion of the original paper is to be used, and if you are not the senior author, also obtain the senior author's approval.

Requirements to be followed when using an entire IEEE copyrighted paper in a thesis:

1) The following IEEE copyright/ credit notice should be placed prominently in the references: © [year of original publication] IEEE. Reprinted, with permission, from [author names, paper title, IEEE publication title, and month/year of publication]
2) Only the accepted version of an IEEE copyrighted paper can be used when posting the paper or your thesis on-line.
3) In placing the thesis on the author's university website, please display the following message in a prominent place on the website: In reference to IEEE copyrighted material which is used with permission in this thesis, the IEEE does not endorse any of [university/educational entity's name goes here]'s products or services. Internal or personal use of this material is permitted. If interested in reprinting/republishing IEEE copyrighted material for advertising or promotional purposes or for creating new collective works for resale or redistribution, please go to http://www.ieee.org/publications_standards/publications/rights/rights_link.html to learn how to obtain a License from RightsLink.

If applicable, University Microfilms and/or ProQuest Library, or the Archives of Canada may supply single copies of the dissertation.

BACK CLOSE WINDOW

CCC RightsLink

Sign in/Register

IEEE Requesting permission to reuse content from an IEEE publication

Active Learning for Node Classification using a Convex Optimization approach

Conference Proceedings: 2022 IEEE Eighth International Conference on Big Data Computing Service and Applications (BigDataService)
Author: Deepesh Agarwal
Publisher: IEEE
Date: August 2022

Thesis / Dissertation Reuse

The IEEE does not require individuals working on a thesis to obtain a formal reuse license, however, you may print out this statement to be used as a permission grant:

Requirements to be followed when using any portion (e.g., figure, graph, table, or textual material) of an IEEE copyrighted paper in a thesis:

1) In the case of textual material (e.g., using short quotes or referring to the work within these papers) users must give full credit to the original source (author, paper, publication) followed by the IEEE copyright line © 2011 IEEE.
2) In the case of illustrations or tabular material, we require that the copyright line © [Year of original publication] IEEE appear prominently with each reprinted figure and/or table.
3) If a substantial portion of the original paper is to be used, and if you are not the senior author, also obtain the senior author's approval.

Requirements to be followed when using an entire IEEE copyrighted paper in a thesis:

1) The following IEEE copyright/ credit notice should be placed prominently in the references: © [year of original publication] IEEE. Reprinted, with permission, from [author names, paper title, IEEE publication title, and month/year of publication]
2) Only the accepted version of an IEEE copyrighted paper can be used when posting the paper or your thesis on-line.
3) In placing the thesis on the author's university website, please display the following message in a prominent place on the website: In reference to IEEE copyrighted material which is used with permission in this thesis, the IEEE does not endorse any of [university/educational entity's name goes here]'s products or services. Internal or personal use of this material is permitted. If interested in reprinting/republishing IEEE copyrighted material for advertising or promotional purposes or for creating new collective works for resale or redistribution, please go to http://www.ieee.org/publications_standards/publications/rights/rights_link.html to learn how to obtain a License from RightsLink.

If applicable, University Microfilms and/or ProQuest Library, or the Archives of Canada may supply single copies of the dissertation.

BACK CLOSE WINDOW

CCC RightsLink

Sign in/Register

A general framework for quantifying aleatoric and epistemic uncertainty in graph neural networks

Author: Sai Munikoti,Deepesh Agarwal,Laya Das,Balasubramaniam Natarajan
Publication: Neurocomputing
Publisher: Elsevier
Date: 7 February 2023

Journal Author Rights

Please note that, as the author of this Elsevier article, you retain the right to include it in a thesis or dissertation, provided it is not published commercially. Permission is not required, but please ensure that you reference the journal as the original source. For more information on this and on your other retained rights, please visit: https://www.elsevier.com/about/our-business/policies/copyright#Author-rights

BACK CLOSE WINDOW

CCC RightsLink

Sign in/Register

Tracking and handling behavioral biases in active learning frameworks

Author: Deepesh Agarwal,Balasubramaniam Natarajan
Publication: Information Sciences
Publisher: Elsevier
Date: September 2023

Journal Author Rights

Please note that, as the author of this Elsevier article, you retain the right to include it in a thesis or dissertation, provided it is not published commercially. Permission is not required, but please ensure that you reference the journal as the original source. For more information on this and on your other retained rights, please visit: https://www.elsevier.com/about/our-business/policies/copyright#Author-rights

BACK CLOSE WINDOW

CCC RightsLink

Sign in/Register

IEEE Requesting permission to reuse content from an IEEE publication

Foundational Models for Fault Diagnosis of Electrical Motors

Conference Proceedings: 2023 IEEE International Conference on Power Electronics, Smart Grid, and Renewable Energy (PESGRE)
Author: Sriram Anbalagan
Publisher: IEEE
Date: 17 December 2023

Thesis / Dissertation Reuse

The IEEE does not require individuals working on a thesis to obtain a formal reuse license, however, you may print out this statement to be used as a permission grant:

Requirements to be followed when using any portion (e.g., figure, graph, table, or textual material) of an IEEE copyrighted paper in a thesis:

1) In the case of textual material (e.g., using short quotes or referring to the work within these papers) users must give full credit to the original source (author, paper, publication) followed by the IEEE copyright line © 2011 IEEE.
2) In the case of illustrations or tabular material, we require that the copyright line © [Year of original publication] IEEE appear prominently with each reprinted figure and/or table.
3) If a substantial portion of the original paper is to be used, and if you are not the senior author, also obtain the senior author's approval.

Requirements to be followed when using an entire IEEE copyrighted paper in a thesis:

1) The following IEEE copyright/ credit notice should be placed prominently in the references: © [year of original publication] IEEE. Reprinted, with permission, from [author names, paper title, IEEE publication title, and month/year of publication]
2) Only the accepted version of an IEEE copyrighted paper can be used when posting the paper or your thesis on-line.
3) In placing the thesis on the author's university website, please display the following message in a prominent place on the website: In reference to IEEE copyrighted material which is used with permission in this thesis, the IEEE does not endorse any of [university/educational entity's name goes here]'s products or services. Internal or personal use of this material is permitted. If interested in reprinting/republishing IEEE copyrighted material for advertising or promotional purposes or for creating new collective works for resale or redistribution, please go to http://www.ieee.org/publications_standards/publications/rights/rights_link.html to learn how to obtain a License from RightsLink.

If applicable, University Microfilms and/or ProQuest Library, or the Archives of Canada may supply single copies of the dissertation.

BACK CLOSE WINDOW