

Maintaining Motivational Value Trade-Offs Through a Values-Oriented
Course Recommendation Tool for Undergraduate Computer Engineering Students

by

Luke Slade

A Thesis submitted to the Faculty of Graduate and Postdoctoral Studies of

The University of Manitoba

In partial fulfilment of the requirements of the degree of

Master of Science

Department of Biosystems Engineering

University of Manitoba

Winnipeg

Copyright © 2026 by Luke Slade

Abstract

Course recommendation tools (CRTs) have been developed to help students navigate through the complex process of course selection. However, many CRTs do not account for the complex varieties of students' course choice motivations. In this study, motivation was viewed through expectancy-value theory (EVT) that describes motivation as a combination of a person's expectations of success (expectancy) and their subjective values regarding the task (attainment, interest, utility, and cost). When choosing courses, students often make trade-offs between course-related factors such as, interesting topics, perceived usefulness, and perceived difficulty. These value trade-offs are used to balance each semester of courses; and this balance is different for each student. When students encounter a largely unbalanced semester, they are at risk of dropping out. As such, the novel CRT developed for this thesis maintains the balance of value trade-offs throughout recommended course pathways. This research project consisted of four sequential phases to develop and evaluate the values-oriented CRT. In Phase 1, a survey was conducted to map engineering students' perceptions of utility and cost onto courses within the University of Manitoba's Computer Engineering program. Then a content analysis was performed to label courses with curricular subjects of interest. In Phase 2, a values-oriented dataset of course pathways was constructed by combining simulated course pathways with the results from Phase 1. In Phase 3 the course recommendation model was designed and implemented as an encoder-based Transformer trained on the dataset developed in Phase 2. Phase 4 consisted of a mixed methods evaluation of the CRT's usability. Usability was examined through the analysis of sensitivity and accuracy of complete course pathways recommendations; and user interviews. The course recommendation model achieved high accuracy and ranking ability for predicting value-maintained semesters of courses. Evaluation results demonstrated

that the CRT is useful for all value trade-offs, and accurately generates pathways aligned with students' expectations. The CRT supports students' motivational values and was perceived as a useful additional support. As such, this thesis demonstrated the feasibility of a novel methodology to develop a CRT that maintains motivational value trade-offs and lays the groundwork for future research on values-oriented CRTs.

Keywords: Course Recommendation Systems, Engineering Education, Expectancy-Value Theory, Human-Centered Artificial Intelligence, Educational Data Mining

Acknowledgements

I would like to thank my family, my partner, and my friends, for their support and encouragement throughout my studies.

I am especially grateful to my advisor Dr. Kari Zacharias for her outstanding support, guidance, and mentorship throughout this research. I am particularly thankful for the trust and freedom she gave me to explore and pursue my research interests. Her thoughtful mentorship has been invaluable to this work and my growth as a researcher.

I also extend my thanks to the participants in my studies for sharing their time and perceptions, as their contributions made this research possible.

Dedication

To my grandma, Alice, for your lifelong love of learning, which you instilled in me. In celebration of your 100th birthday, thank you for all your years of love, wisdom, and inspiration.

Table of Contents

Abstract	i
Acknowledgements	iii
Dedication	iv
Table of Contents	v
List of Tables	vii
List of Figures	ix
Chapter 1: Introduction	1
1.1. Research Purpose and Objectives	4
1.2. Research Site	5
1.3. Study Design	6
1.4. Positionality	8
1.5. Document Structure	9
Chapter 2: Literature Review	10
2.1. Course Selection Motivations	10
2.2. Course Recommendation Tools	13
2.3. Machine Learning Architectures for Course Recommendation	15
2.3.1. Early Machine Learning-Based Course Recommendation Approaches	16
2.3.2. The Transformer Architecture for Course Recommendation	17
2.4. Values-Oriented Design Approaches	21
2.5. Value Framework: Eccles' Expectancy-Value Theory of Achievement Motivation	27
2.5.1. EVT in Engineering Education Research	29
2.5.2. Integrating EVT into Course Recommendations	33
Chapter 3: Value Mapping	36
3.1. Survey of Students' Course Values Perceptions	37
3.2. Survey Validation	39
3.3. Survey Results	42
3.3.1. Examples of Maintaining Course Values in the Preliminary Year	51
3.4. Course Topic Labelling	57
3.5. Course Topic Labelling Results	58
Chapter 4: Dataset Construction	62
4.1. Simulating Synthetic Course Pathways	63
4.2. Dataset Development	65
4.3. Resulting Datasets	74

Chapter 5: Design and Implementation of my Course Recommendation Model.....	77
5.1. Tool Architecture	78
5.1.1. Transformer Model Overview	78
5.2. Designing Training Examples for the Transformer Model	81
5.2.1. Characteristics of Transformer Training Data.....	82
5.2.2. Representation of Courses Completed.....	83
5.2.3. Representation of Student Value Trade-Offs	84
5.2.4. Model Input Formats.....	88
5.3. Loss Function.....	89
5.4. Training Metrics.....	91
5.5. Initial Training Pseudo Validation on Small Datasets.....	93
5.6. Model Optimization and Selection	94
5.6.1. Hyperparameter Optimization	95
5.6.2. Optuna.....	96
5.6.3. Broad Initial Hyperparameter Search Space.....	98
5.6.4. Refined Hyperparameter Search Spaces	99
5.6.5. Comparison of Optimized Models.....	99
5.6.6. Model Selection Using K-Fold Cross Validation.....	100
5.7. Machine Learning Model Results.....	101
5.8. Cross Validation Results	102
5.9. Best ML Model Output Examples	105
5.10. CRT Integration.....	108
Chapter 6: Evaluation	114
6.1. Sensitivity Analysis.....	117
6.1.1. Sobol Indices.....	119
6.1.2. Delta Moment-Independent Indices.....	119
6.1.3. Input Distribution Analysis	120
6.1.4. Value Sensitivity Criteria	122
6.2. Pathway Value Accuracy Analysis.....	123
6.2.1. Value Accuracy Criteria	124
6.3. Semi-Structured User Interviews	126
6.4. Evaluation Results	128
6.4.1. Sensitivity Results.....	129
6.4.2. Pathway Maintained Values Accuracy Results	132

6.4.3. Interview Findings	136
6.5. Evaluation of my Piloted Values-Oriented Course Recommendation Tool.....	144
Chapter 7: Discussion	147
7.1. Contributions.....	150
7.2. Limitations	152
7.3. Future Directions	153
7.4. Conclusions.....	154
7.5. Learning Reflection	158
Ethics Statement.....	160
References.....	161
Appendix A: UM Computer Engineering Flowchart 2025 – 2026.....	168
Appendix B: Consent Forms.....	169
Appendix C: Cronbach's Alpha for Survey	182
Appendix D: Synthetic Course Pathways for Dataset	183
Appendix E: Hyperparameter Optimization Results and Training Behaviour	184
Appendix F: Evaluation Sensitivity Plots	211

List of Tables

Table 1. EVT values and representative questions	28
Table 2. Value perceptions survey statements.....	38
Table 3. Survey item intercorrelation.....	40
Table 4. Survey item factor loading.....	41
Table 5. Survey quantization scale.....	43
Table 6. Statistics of course utility and cost representations	44
Table 7. Cronbach's alpha of utility for courses with at least N responses	45
Table 8. Cronbach's alpha for cost for courses with at least N responses.....	46
Table 9. Descending value rankings excluding electives, $N \geq 5$	47
Table 10. Descending value rankings of grouped electives, $N \geq 5$	48
Table 11. Descending value representations of preliminary-year courses, $N \geq 9$	54
Table 12. Utility maintained example of preliminary year pairings	56
Table 13. Cost maintained example of preliminary year pairings	56
Table 14. Utility minus cost maintained preliminary year pairings.....	56
Table 15. Interest coding results	59
Table 16. Numerical course representation of ECE 2400	61
Table 17. Maintained value mean and std. dev. for the synthetic course pathways tree.....	68
Table 18. Statistics of interest topics not occurring in semesters	69

Table 19. 4-year course pathway example	72
Table 20. Pathway example semester-based utility and cost averages	72
Table 21. Pathway example maintained utility and cost binning.....	73
Table 22. Pathway example semester interest max scaling	73
Table 23. Pathway example maintained interest binning.....	73
Table 24. Expectancy labelling for time to graduation and enrollment maximum.....	73
Table 25. Total dataset size resulting from all synthetic pathways for each input format	75
Table 26. Sizes of datasets used for model optimization and selection	75
Table 27. Initial hyperparameter search space	99
Table 28. Summary of best models.....	103
Table 29. Maintained value trade-offs input and output	113
Table 30. CRT input parameters	118
Table 31. Input distributions for accuracy analysis	121
Table 32. Performance levels for value sensitivity	122
Table 33. Performance levels for input accuracy.....	124
Table 34. Performance levels for likelihood of accuracy	125
Table 35. Performance levels of extra semesters when inaccurate expectancy	125
Table 36. Average parameter sensitivity ranks under each EVT construct accuracy metric (lower is more sensitive; with ties).....	129
Table 37. Mean binary accuracy for each maintained value bin across values (weighted equally) by distribution (higher is more accurate).....	131
Table 38. Mean binary accuracy for each maintained value bin across values by semesters.....	131
Table 39. Maintained accuracy performance likelihood results	133
Table 40. Likelihood of maintained value improvement from “average” to “more” when inaccurate for “average”	134
Table 41. Expectancy accuracy performance likelihood results	135
Table 42. Cronbach's alpha for utility	182
Table 43. Cronbach's alpha for cost	182
Table 44. Varying width simulated pathways	183
Table 45. Static width simulated pathways.....	183
Table 46. Combined simulated pathways	183
Table 47. Top 3 CTV HPO models from the broad search (round 1)	185
Table 48. Top 3 Semester Sequence HPO models from the broad search (round 1)	187
Table 49. Top 3 Semester Sequence HPO models from the refined search (round 2).....	190
Table 50. Change in performance metrics of top 3 Semester Sequence models from HPO round 1 to round 2	192
Table 51. Top 3 Hybrid HPO models from the broad search (round 1).....	193
Table 52. Top 3 Hybrid HPO models from the refined search (round 2)	195
Table 53. Change in performance metrics of top 3 Hybrid models from round 1 to round 2	198
Table 54. Configurations of top HPO Semester Sequence models from each round of HPO	201
Table 55. Metric results of top HPO Semester Sequence models from each round of HPO.....	202
Table 56. Hyperparameter configurations of top HPO hybrid models from each round.....	205
Table 57. Performance metrics of top HPO hybrid models from each round of HPO	205
Table 58. Refined search space for the Semester Sequence model	209
Table 59. Refined search space for the Hybrid model.....	209
Table 60. HPO round 1 timing.....	210

Table 61. HPO round 2 timing	210
Table 62. Training times of best models	210
Table 63. 5-Fold cross validation timing	210

List of Figures

Figure 1. Four-phase study design to develop and evaluate the tool	6
Figure 2. Transformer encoder attention and feed forward network	19
Figure 3. EVT applied to course choice motivation	28
Figure 4. Phase 1: Value Mapping	36
Figure 5. How many courses had a number of responses	42
Figure 6. Percent of courses included when $N \geq$ the number of responses	43
Figure 7. Combined utility and cost for each course by year and electives.....	50
Figure 8. Coded ECE 2400 syllabus sections with highlighted topics	60
Figure 9. Phase 2: Dataset Construction	62
Figure 10. From pathways to a tree of unique sub-pathways	67
Figure 11. Maintained value binning estimated by a normal distribution	70
Figure 12. Phase 3: Course Recommendation Model Implementation	77
Figure 13. N-layer multi-head attention encoder-only Transformer architecture	80
Figure 14. Example model output without a course history	80
Figure 15. Input representations of courses taken	84
Figure 16. Categorical maintained value vectors.....	85
Figure 17. Concatenated 1 x 9 utility input vector.....	86
Figure 18. Input vectors space for motivational value aspects	87
Figure 19. Model input structures including EVT preferences and completed courses	88
Figure 20. Model optimization and selection process	94
Figure 21. Machine learning model selection process and results.....	101
Figure 22. Cross validation portion of the model selection process	102
Figure 23. 5-fold cross validation mean (line) and 1 standard deviation (area) for Semester Sequence (left) and Hybrid (right) models	104
Figure 24. Utility and cost trade-off prediction comparison for semester 1	106
Figure 25. Output probability distribution comparison between first and third semester	107
Figure 26. Output probability distribution comparison between fifth and seventh semester	108
Figure 27. Input example with course history and value trade-offs.....	110
Figure 28. Complete course pathway recommended from the input example	111
Figure 29. Regenerated complete course pathway recommended from the input example	112
Figure 30. Phase 4: CRT Evaluation	114
Figure 31. Hyperparameter optimization portion of the implementation process.	184
Figure 32. CTV model HPO round 1 search space coverage and accuracy	185
Figure 33. CTV HPO round 1 accuracy convergence	186
Figure 34. Semester Sequence model HPO search space coverage and accuracy.....	188
Figure 35. Semester Sequence HPO round 1 accuracy convergence	189
Figure 36. Semester Sequence model HPO round 2 search space coverage and accuracy	191
Figure 37. Semester Sequence HPO round 2 accuracy convergence	192

Figure 38. Hybrid model HPO round 1 search space coverage and accuracy	194
Figure 39. Hybrid HPO round 1 accuracy convergence	195
Figure 40. Hybrid model HPO round 2 search space coverage and accuracy	197
Figure 41. Hybrid HPO round 2 accuracy convergence	197
Figure 42. Training behaviour analysis portion of the model selection process.....	199
Figure 43. Optimized CTV model training behaviour on 1% of dataset for 30 epochs	201
Figure 44. Comparison of training behaviour of the top Semester Sequence HPO models (round 1 left, round 2 right) on 1% of the dataset	203
Figure 45. Comparison of training behaviour of the top Semester Sequence HPO models (round 1 left, round 2 right) on 2% of the dataset	204
Figure 46. Comparison of training behaviour of the top Hybrid HPO models (round 1 left, round 2 right) on 1% of the dataset	207
Figure 47. Comparison of training behaviour of the top Hybrid HPO models (round 1 left, round 2 right) on 2% of the dataset	208
Figure 48. Sobol analysis attainment plots	211
Figure 49. Sobol analysis utility plots.....	212
Figure 50. Sobol analysis cost plots.....	213
Figure 51. Sobol analysis interest plots	214
Figure 52. Delta analysis attainment plots.....	215
Figure 53. Delta analysis utility plots	216
Figure 54. Delta analysis cost plots	217
Figure 55. Delta analysis interest plots	218

Chapter 1: Introduction

Undergraduate students' course selection is a crucial contributor to their experiences and achievement at university; however, the course planning process can be complicated. The complexity of course planning stems from the many factors (personal and institutional) that are involved in the process that often leads to overwhelming and messy student experiences despite having access to academic supports (Babad and Tayeb 2003; Chen et al. 2022). When students struggle to navigate through course selection, potential impacts include dropping out of their program or being unable to graduate on time (Borchers and Pardos 2023; Molontay et al. 2020).

When planning their courses, students are motivated by different factors. Ma et al. (2021) clustered students with similar motivations into four distinct groups of course planning (Ma et al. 2021). These groups were:

1. grade-oriented students who aimed to take low difficulty courses to maximize their GPA,
2. learning-oriented students who chose courses based on their interests and their perceived usefulness of courses,
3. social-oriented students who chose courses to join their friends, and
4. balance-oriented students who made trade-offs between course difficulty, usefulness, and interests to find a personal balance.

Ma et al.'s (2021) course selection orientations highlighted a variety of approaches to course selection. The variations are further emphasized by the orientations deriving from data clusters, implying that students of the same orientation still have their own personal approach, which may be slightly different from others within the same cluster. Within engineering education research, Eccles' expectancy-value theory (EVT) of achievement motivation (Wigfield and

Eccles 2000) is widely used to describe engineering students' motivations behind their academic decisions. However, little research has explored engineering students' course choice motivations. A potential reason for this lack of literature may be due to the rigidity of engineering programs structured to meet accreditation requirements. To meet accreditation requirements, engineering programs are prescriptive in course requirements, which may provide little flexibility for students aiming to graduate in four years of studies. As such, supports designed to aid students' navigation through course selection often lack personalization based on students' course choice behaviours and motivations.

To support students' navigation through course planning experiences, researchers have developed course recommendation tools (CRTs). CRTs have the potential to aid in reducing students' time and effort when selecting courses and can assist in identifying if students' need to meet with an academic advisor for further consultation. Unsuccessfully navigating through course choices can impact graduating on time, early career opportunities, and persistence. As course selection and its potential negative impacts are complex, it is important for CRTs to account for such complexities through personalization to be effective supports (Esteban et al. 2018; Ma et al. 2021). However, much CRT research fails to adequately represent the variety of students' motivations.

Modern CRTs represent students' behaviours using machine learning (ML) models trained on historical data to make recommendations. By training on historical data, there is a risk of perpetuating biases and failing to discover uncommon recommendations that may lead to better outcomes (Pardos and Jiang 2020). Course recommendation models commonly focus on predicting students' GPA performance to maximize grades, however other foci have been used such as finding unexpected yet useful courses to provide alternatives (Chen et al. 2022; Pardos and Jiang 2020); analyzing workloads to reduce dropout (Borchers and Pardos 2023); or

recommending complete course pathways built around user selected courses (Shao et al. 2021).

Although these CRTs have been useful for some students, each approach used historical data and failed to consider the complexity of motivational factors driving course selection – both of which are common oversights within CRT literature.

Recent approaches to support personalized educational experiences for engineering students have looked towards popular large language model (LLM) tools such as ChatGPT (OpenAI 2024), Gemini (Team Gemini 2025), and DeepSeek (DeepSeek-AI 2024), to provide personalized explanations, adaptive feedback, and virtual tutoring (Harris and Kittur 2024; Qadir 2023).

Studies in engineering education have reported that students and instructors perceive LLM tools as valuable for improving productivity and supporting learning processes, while also providing flexible and accessible learning experiences (Fatahi et al. 2023; Guillén-Yparrea and Hernández-Rodríguez 2024; Harris and Kittur 2024). The underlying machine learning architectures of LLMs have also been used to develop CRTs for undergraduate students (Moon et al. 2024; Shao et al. 2021); however, little research has been conducted on these tools.

To address these gaps, I sought to develop a CRT for engineering students that effectively models the many motivating factors that drive students' course planning. As this context and approach to CRT development is novel, there is no established methodology for building such tools, and key gaps remain in the literature on CRTs. These gaps include:

- 1) Limited research on ML modelling of many factor course choices,
- 2) Little research on engineering-specific CRTs and,
- 3) Lack of systematic integration of students' motivational values into course recommendation models.

To address these three gaps, I investigated values-oriented design methodologies that blend social science theories and machine learning to develop a novel methodology for constructing a values-oriented CRT. As machine learning consists of modeling the underlying information within a dataset, my research focused on the systematic integration of engineering students' motivating course choice factors into a dataset that supports course recommendations. To evaluate the feasibility of my methodology, I evaluated the performance and usability of a CRT trained on my constructed dataset.

1.1. Research Purpose and Objectives

The purpose of this research was to develop and examine the feasibility of a novel approach for constructing a motivation-focused and values-oriented dataset of course pathways for academic CRTs. My approach systematically modelled and integrated many course selection motivational values of undergraduate computer engineering students into a machine learning dataset. To evaluate the feasibility of my approach, I developed a course recommendation tool using my values-oriented dataset and then used mixed methods to evaluate the usability of the tool.

The objectives of this research were as follows:

1. Develop and pilot a methodology of systematically integrating motivational values into a course recommendation tool for undergraduate computer engineering students at the University of Manitoba.
2. Evaluate the feasibility of the piloted methodology.

1.2. Research Site

The research site for this study was the University of Manitoba (UM), and the research focused specifically on the undergraduate program in Computer Engineering. All Canadian engineering programs are accredited by the Canadian Engineering Accreditation Board, however, institutions provide varying contexts (supports, course registration policies, course offerings) surrounding their engineering programs that affects engineering students' course pathway planning experiences. At the University of Manitoba, undergraduate engineering students pay tuition based on the number of courses taken each semester, on top of continuing fees, and are given flexibility in the order in which mandatory courses are completed. This flexibility leads students to planning their own course pathways and often extending the length of their program past the expected completion time of 4 years by taking fewer courses in each semester. Engineering programs at other institutions have less flexibility for students to create their own course pathways. Additionally, the academic supports at UM do not include a CRT. As such, UM's flexible engineering programs provide a space where a CRT can be more useful to engineering students compared to other institutions' engineering programs. At the UM, there are five engineering programs. I am most familiar with UM's computer engineering courses and academic supports when compared to the other programs since I am a recent graduate of the UM Computer Engineering program. As such, the UM Computer Engineering program was an appropriate site for this methodological exploration and pilot tool development.

1.3. Study Design

My research project followed a four-phase approach. Phases 1 and 2 of my study combined course choice behaviours and motivation theory, engineering students' course choice motivation values, and synthetic course pathways to construct a values-oriented machine learning dataset for course recommendations. Phases 3 and 4 of my study were the implementation of a CRT by developing a machine learning model and subsequently evaluating the CRT using mixed methods. An overview of my study is presented in Figure 1.

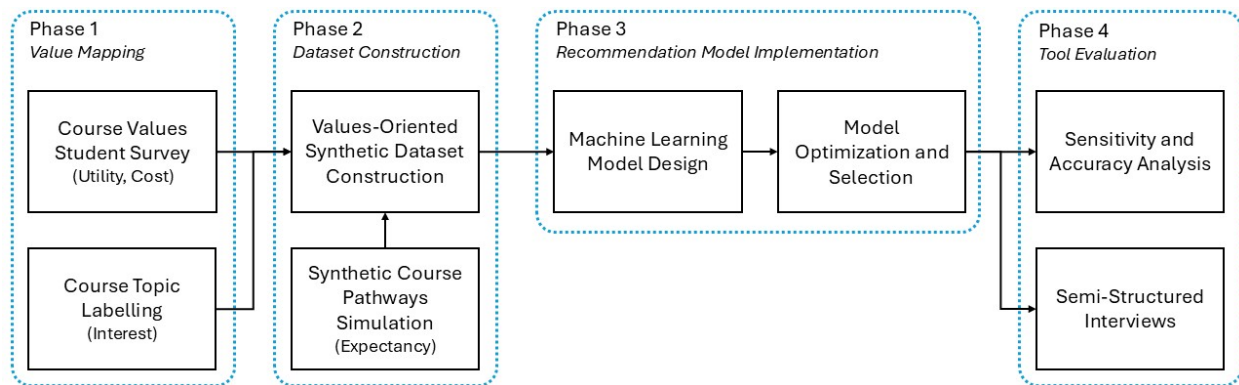


Figure 1. Four-phase study design to develop and evaluate the tool

In Phase 1 of the project, I used EVT to map values onto the courses in the computer engineering program, based on students' perceptions of those courses. I conducted a survey of computer engineering students' course value perceptions, and subsequently used course topic labelling to map the EVT values of utility, cost, and interest onto courses in the University of Manitoba's computer engineering program. The collection of the values data supported the development of a dataset in Phase 2.

In Phase 2, I developed a dataset of values-oriented course pathways based on the value mapping. First, I modelled the UM computer engineering program and simulated course pathways to graduation. The simulated course pathways allowed me to construct a dataset of unique synthetic course pathways, avoiding privacy issues that could result from using course pathways of real engineering students to train an ML model. I then used the results from Phase 1, combined with theory on course choice orientations and maintained motivational values, to categorize each synthetic course pathway based on the maintained amount of EVT values across semesters in each pathway. These categories enabled learnable patterns to emerge that connected course pathways to EVT value trade-offs. As such, this dataset supported the development and training of an ML model in Phase 3.

In Phase 3, I designed and implemented a machine learning model for course recommendation based on the dataset developed in Phase 2. The machine learning architecture I implemented was based on the Transformer (Vaswani et al. 2017) that is the backbone for state-of-the-art sequence prediction models such as course recommendation (Moon et al. 2024; Shao et al. 2021). As such, I formatted my values-oriented dataset of course pathways into labelled training examples (supervised learning) as sequential data to align with the Transformer architecture. I investigated three training approaches to representing a user's course history as a program completion checklist, academic transcript, and a combination of both. After I found the highest performing model, I moved on to Phase 4 to evaluate its performance.

Finally, in Phase 4, I evaluated my CRT using mixed methods for a holistic evaluation of its feasibility through the lens of usability. I used sensitivity analyses, an accuracy analysis, and semi-structured user interviews to examine the usability of the tool and identify future design directions. Sensitivity analyses examined the quality of the connection between inputted

maintained value trade-offs and their corresponding maintained values within course pathway recommendations, as well as the effects of different input distributions on the accuracy of the tool. Next, I analyzed the accuracy of the tool for each EVT value and each category of maintained value. Then, I conducted semi-structured interviews of UM Computer Engineering students for their perceptions of my CRT. I used the sensitivity results, accuracy results, and interview findings to examine the usability of my CRT. I also used the interview findings to investigate engineering students' suggestions to improve my CRT as an academic support for course planning.

1.4. Positionality

I approached this work as an alumnus from the UM computer engineering program (the degree program I developed this CRT for). As a recent graduate, I understood what the program was like during my time as an undergraduate student, which helped me understand what it is like to choose courses and plan course pathways. At the same time, as a domestic student from a middle-class family and as part of the visual majority, I also recognized my own biases in the way I viewed courses (usefulness and difficulty), course selection (registration and fees), and pathway planning (workload and graduation expectations), which stemmed from my own motivational values and my overall approach to this study. While this study was motivated in part by my own experiences, it was therefore important to ground the work in the perceptions of other Computer Engineering students, in terms of the value representations of courses (and consequently the machine learning training data) and in the evaluation of my CRT. Grounding the data for this study in the perceptions of other computer engineering students reduced bias and increased the representativeness of this research. The approach of my research centered the CRT

around the voices and values of current UM Computer Engineering students and their course selection experiences, in addition to my own.

1.5. Document Structure

Following this introductory chapter, Chapter 2 presents my thematic literature review. Chapter 3 to Chapter 6 present Phases 1 to 4 of my study. The methodology and results from each phase are presented sequentially where each chapter presents one Phase. As such, Chapter 3 presents the value mapping, Chapter 4 presents my construction of the values-oriented dataset, Chapter 5 presents my design and implementation of my course recommendation model, and Chapter 6 presents my mixed methods evaluation of my CRT. Chapter 7 includes my discussion of my research, the contributions my research makes, and my conclusions regarding this research project.

Chapter 2: Literature Review

The design of my values-oriented course recommendation tool for undergraduate computer engineering students was informed by literature from five interconnected areas: education, course recommendation tools, machine learning, values-oriented design, and expectancy-value theory. First, I explore education research on the motivations behind students' course selection, which helped me define the conceptual framework for my design. Second, I review studies and designs of previous course recommendation tools (CRTs) to understand the current state of the field and identify gaps in existing practice. Third, I discuss machine learning (ML) approaches as the most common approaches to implementation of modern CRTs with a particular focus on the Transformer architecture. I argue that the Transformer architecture is currently the most promising ML approach for course recommendation. Fourth, I explore values-oriented design frameworks for computer systems to draw inspiration from for the design of my research, and the integration of values into my CRT. Finally, I review Eccles' expectancy-value theory (EVT) of achievement motivation as the theoretical value framework for my research, discuss how EVT is situated within the field of engineering education, and explain how I operationalized EVT for computer engineering students' course choice values.

2.1. Course Selection Motivations

Before a course recommendation tool can be developed, it is necessary to understand what motivates students to select their courses in the first place. The nature of motivation has been described as “energy, direction, persistence and equifinality – all aspects of activation and intention” (Ryan and Deci 2000). In other words, actions can be performed with different amount

of energy, potentially with a goal or direction in mind, may require persisting through time, and any combination of these can lead to the same result of completing the intended action.

Equifinality (reaching the same end with various means) is a key concept of motivation since different factors can motivate different people to complete the same task. Additionally, equifinality is important for CRTs as students reach the same goal (graduation) through many different course pathways.

Theories such as self-determination theory (SDT) and expectancy-value theory (EVT) have been used to explain the motivations behind peoples' actions (Ryan and Deci 2000; Wigfield and Eccles 2000). SDT and EVT are complementary theories. SDT focuses more on the sources of motivation, whereas EVT focuses on describing motivation in more depth to specifically describe peoples' expectations of success and their value trade-offs that, together, affect the decision to do or not do a task. Ryan and Deci (2000) used SDT to divide motivation into intrinsic and extrinsic motivation and argued that intrinsic motivation is more powerful than extrinsic motivation. In Ryan and Deci's (2000) research, intrinsic motivation was found to produce more interest, excitement, and confidence (compared to extrinsic motivation), which enhanced performance, persistence, self-esteem, and general well-being. EVT argues that actions of choice and persistence can be explained by peoples' beliefs about their success alongside their value trade-offs (Wigfield and Eccles 2000). Within Wigfield and Eccles' research on EVT, motivation was represented in five constructs including expectancy, attainment, interest, utility, and cost, the later four being subjective values. Depending on the action, the EVT values contribute at various levels with some values being more important to certain actions than others. Trade-offs often occur between the four EVT values when motivating choice. For example, a high-cost task might be outweighed by the usefulness or enjoyment of the task, which results in

motivation to perform the task. Conversely, the high cost of a task with little usefulness or importance would lead to a lack of motivation.

SDT and EVT are both common motivational theories used in education research for explaining the motivations of students and the various actions or task they perform (Kirkham et al. 2023; Lee et al. 2021; Pintrich 2003), however EVT is prevalent in engineering education research (see Chapter 2.5 below). In a study that used SDT and EVT to investigate the reasons undergraduate students enroll in courses, students of various majors were found to choose courses for extrinsic reasons and based on task values rather than expectations of success (Lee et al. 2021). However, Lee et al. (2021) also found that students' expectations of interest and learning career related skills were more often used as guiding factors for choosing courses thought to be important. Courses thought to be difficult were chosen more often for self-development or as a program requirement.

The process of course selection undergone by many general undergraduate students has been described as “overwhelming” (Chen et al. 2022) and “quite messy and disorganized” (Babad and Tayeb 2003). However, research on course choice motivation can help to make sense of this apparent mess. A survey of general undergraduate students found guiding factors for course selection and clustered students into four different groups according to which factors they prioritized when choosing courses (Ma et al. 2021). These four groups were balance, grades, learning, and socializing. Balance-oriented students focused on balancing workload and making trade-offs between perceived course difficulty, usefulness, and interest. Grades-oriented students prioritized their grades above other factors and tended to avoid courses perceived as difficult or time-consuming in favour of choosing courses that they believed would result in a high grade. Learning-oriented students cared less about grades and instead chose courses they perceived as

interesting or useful. Social-oriented students chose courses based on recommendations from peers and to spend time with their friends in class.

To develop a values-oriented CRT, it is important to understand students' course choice behaviours and connect those behaviours to motivational values. Of the four approaches to course selection identified by Ma et al. (2021) and the two motivational theories, the most aligned pairing is the balance-oriented approach and EVT. The balance-oriented approach to course selection focuses on trade-offs between course difficulty, usefulness, and student interests and is parallel to how EVT describes motivation as value trade-offs between cost, utility, and interest. As such, my conceptual framework for describing values-oriented course choices was formed through the combination of Ma et al.'s (2021) balance-oriented approach and EVT. The next section reviews CRTs as an academic advising support and reviews how students' values are represented within existing CRT research.

2.2. Course Recommendation Tools

Course recommendation tools are one of the many supports that may be available to undergraduate students. In most cases, however, academic advisors are still the primary source of course selection support for students. One US-based study showed that 73% of undergraduate students across 209 institutions in the United States met with an academic advisor 1 to 3 times in their first year of study (Fosnacht et al. 2017). With the high demand for academic advising, academic advisors can be overburdened due to a large ratio of students to advisors (Algarni and Sheldon 2023; Khali and Williamson 2014). In a study of academic advising for engineering students (Khali and Williamson 2014), students were found to often show up at the academic

advisor's office without booking an appointment and some of such students did not seek support until they were close to dropping out. In addition, engineering students felt ignored if responses to emails or phone calls took too long. The engineering academic advisor aided students with a variety of issues such as career choice, course planning, registration, and conflicts with instructors. The most prominent issues for engineering students were related to course planning that accounted for 46% of all advising support provided. Academic course recommendation systems aim to support both academic advisors and students by quickly presenting meaningful information related to planning course pathways (Algarni and Sheldon 2023; Atalla et al. 2023).

Course recommendation tools aim to meaningfully recommend courses to students based on the courses students have previously completed alongside a variety of additional information. To personalize recommendations for each student, these tools have commonly integrated course grades as part of the input to the tool with the goal of maximizing total degree GPA (Atalla et al. 2023; Morsy and Karypis 2018; Pardos et al. 2018). Other approaches have looked beyond maximizing GPA; aiming instead to generate unexpected but useful recommendations (Chen et al. 2022; Pardos and Jiang 2020), predict semester workload (Borchers and Pardos 2023), or generate course pathways based on user-selected courses of interest across future semesters (Shao et al. 2021). Such CRTs only enabled a single approach to course selection and lack opportunities for students to make trade-offs more closely resembling how students choose courses.

Existing studies also rarely reflect a wide range of student characteristics and preferences (Algarni and Sheldon 2023). Some researchers have begun to take participatory (student involved) approaches to the development and evaluation of CRTs (Chen et al. 2022; Shao et al. 2021), however a CRT explicitly created with and for engineering students has yet to be

explored. As such, the CRT developed for my thesis addressed these gaps by combining a variety of goals through the trade-offs associated with a balance-oriented approach to course selection (Ma et al. 2021) and the EVT value framework (Wigfield and Eccles 2000). Additionally, my research followed a recommendation of Algarni and Sheldon's (2023) literature review to use a sophisticated machine learning model to improve the memory and accuracy of the CRT. In the next section of my literature review I discuss the types of machine learning models used to develop course recommendation tools and how the recently popularized Transformer architecture (Vaswani et al. 2017) enables both improved memory and accuracy.

2.3. Machine Learning Architectures for Course Recommendation

Machine learning is the most common approach to developing modern pathway advising tools (Algarni and Sheldon 2023; Cui et al. 2019; Guruge et al. 2021; McConvey et al. 2023).

Recommending a course is often structured as a sequence prediction problem where a sequence of courses is input, and a course is output. Machine learning is often applied to sequence prediction problems as it can learn complex relationships of sequential data. Early course recommendation tools were based on the course2vec model (Pardos et al. 2018) that enabled structured course representations, while newer approaches (Jiawei Li et al. 2025; Moon et al. 2024; Shao et al. 2021) leveraged the recent major advancement in sequence prediction by using a Transformer model (Vaswani et al. 2017).

2.3.1. Early Machine Learning-Based Course Recommendation Approaches

A student's course pathway contains context based on the ordering of courses, similar to how the words in a sentence contain context based on their order. The course2vec machine learning architecture, which was the basis for most early ML-based approaches to course recommendation, built on the idea of text prediction to predict a student's next course based on the courses they have previously taken (Pardos et al. 2018). The course2vec model, based on the skip-gram word2vec embedding model (Mikolov et al. 2013), mapped courses to a high dimensional vector space. In the word2vec model, the higher dimensional space facilitated meaningful relationships between words, e.g. aunt and uncle are related in a similar manner as mother and father. Higher dimensional space allowed for different dimensions to represent different relational aspects of words. In the course2vec model, meaningful relationships between courses were learned using a high dimensional vector. The course2vec model was trained on historical enrollment data to recommend a course based on the user's course history. The course2vec model was effective at creating meaningful representations of similar courses and clustering historical enrollments by major.

Researchers have created pathway advising models by expanding on the course2vec architecture (Morsy and Karypis 2018; Pardos and Jiang 2020). Pardos and Jiang (2020) developed serendipitous pathways that were meant to be unexpected but successful. Morsy and Karypis (2018) used the course2vec architecture to create pathways that improve GPA and graduation time. Both of these CRTs resulted in successful course recommendations that met their design goals. The course2vec and skip-gram architectures has also been used in other ways. Instead of using course2vec to make course recommendations, Borchers and Pardos (2023) used

course2vec to create additional input features to enhance seven other models predict the workload of courses based on time load, mental effort, and psychological stress.

The limitations of the course2vec model arose from its lack of long-term memory (largely recommending based of a few recent courses) and reliance on past enrollment data for training.

The course2vec model learned recommendations from what courses were in close proximity within students' enrollment history, meaning that the model did not learn prerequisite structures or the underlying motivations behind students' decisions to enroll in a course. Course2vec's reliance on past enrollment reduced recommendation quality for students with few or no courses previously completed since there was little information to base recommendations on.

Additionally, by training on past enrollment data course2vec risked perpetuating biases through a filter-bubble as recommendations were based on what enrollment order other students followed (Pardos and Jiang 2020). Such biases may lead to recommending course pathways with high-workload semesters or increased time within the program by initially enrolling in an unlucky set of courses. Although course2vec approaches have seen success, a new wave of course recommendation systems has begun to leverage newer machine learning models for improved recommendation accuracy and curricular modelling.

2.3.2. The Transformer Architecture for Course Recommendation

The Transformer architecture (Vaswani et al. 2017) has shown improvements in course pathway recommendation compared to the course2vec architecture (Shao et al. 2021). The Transformer architecture is the backbone of all modern large language models (LLMs) such as ChatGPT (OpenAI 2024), Gemini (Team Gemini 2025), and DeepSeek (DeepSeek-AI 2024). Transformer

models excel at predicting what comes next in a sequence, and have also demonstrated effective applications for machine translation (e.g. English to French), document generation (e.g. filling a report template with data from a paragraph of text) and syntactic parsing (e.g. grammatical analysis of text), which go beyond the basics of text generation (e.g. predicting the next word in a sentence) (Vaswani et al. 2017). The effectiveness of Transformer stems from extracting meaningful context from inputs and retaining information from the training data more effectively than previous architectures (Radford et al. 2018; Vaswani et al. 2017).

For course recommendation, Shao et al. (2021) adapted the Transformer architecture with the goal of building course pathways surrounding an input of a few chosen courses and which semesters to enroll in those courses. Results from the Transformer-based CRT (Shao et al. 2021) showed improvements in Recall@10 (accuracy) compared to previous architectures trained on the same data, successful multi-semester planning, and strong recommendation performance for students with few or no previous courses taken.

Shao et al. (2021) implemented the Transformer architecture as a Bidirectional Encoder Representations from Transformers (BERT) model. Training data for the model was constructed from course enrollments and unique courses offered between 2008 and 2019 from 19 higher education institutions in the United States. Shao et al.'s (2021) CRT was compared against a variety of machine learning models such as K-Nearest Neighbors (KNN), Long Short-Term Memory (LSTM), and a standard BERT model. Results from Shao et al.'s (2021) study showed that students found the generated course pathways to be sensible and specific. Other BERT-based models have also seen success when used to recommend massive open online courses (Moon et al. 2024), and to investigate transparency improvements for course recommendations using an

LLM (Jiawei Li et al. 2025). Figure 2 displays the inner components of the encoder representation of a Transformer as used in a generic BERT model.

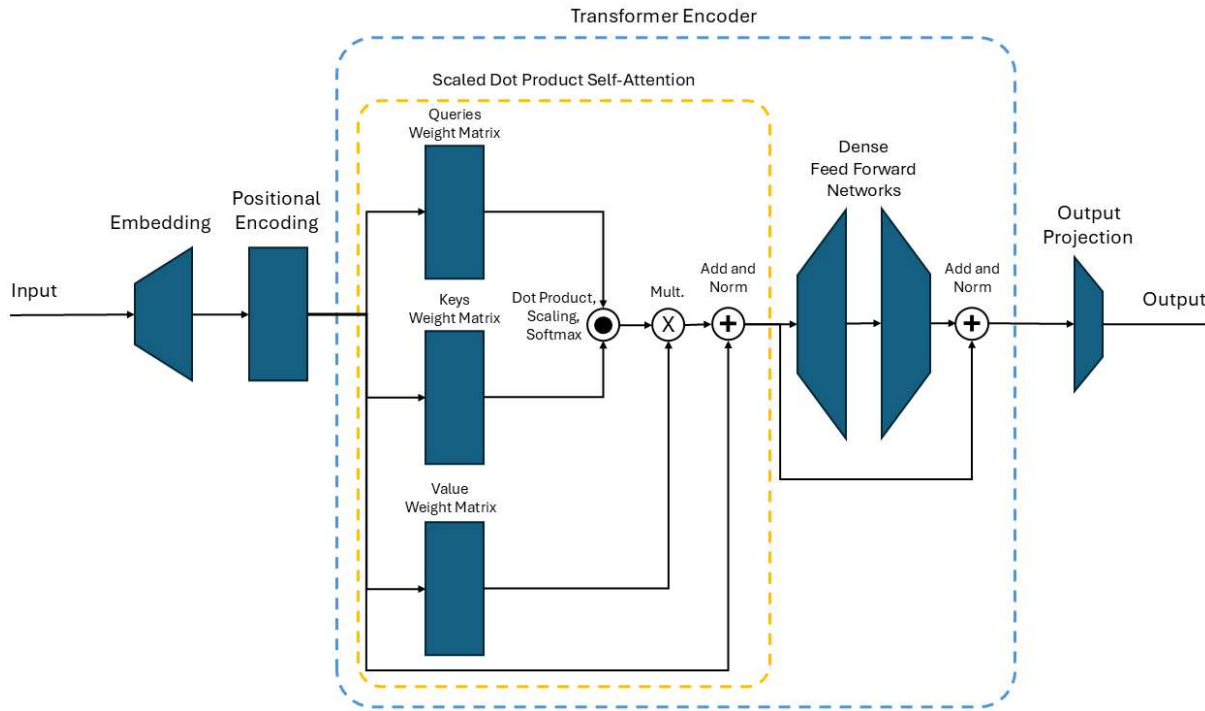


Figure 2. Transformer encoder attention and feed forward network

Based on the underlying components of the Transformer encoder architecture, the model has the potential to tackle the recommendation for improved accuracy and memory for ML-based course recommendation systems (Algarni and Sheldon 2023). The attention-mechanisms (learned weight matrices for extracting input context) inside Transformers were an improvement from previous architectures by effectively extracting meaningful context from input sequences (Radford et al. 2018). First, the 3 weight matrices, Query, Key, and Value, are each multiplied by the embedded input sequence. Then the dot product of the Query and transposed Key matrix was

scaled and passed through a softmax activation before being multiplied by the Value matrix and passed through the feed forward networks. Although the Value matrix uses the term “Value”, it is not related to motivational values or other personal values. Within the attention-mechanisms, each element of the input sequence attends to (is connected to) each other element in the sequence. Attention-mechanisms improved upon recurrent neural network (RNN) models where input elements only affect other nearby inputs. In RNN models the distance between elements within a sequence affects the strength of the relation between them, limiting context extraction, whereas attention-mechanisms allow each element to interact with other elements regardless of the distance between them. In the research field of natural language processing, the attention-mechanism is very effective because it allowed words that have a large impact on the context of a paragraph but appear, for example, at the very beginning of the paragraph, to properly effect the prediction of the next words following the paragraph.

The dense feed forward networks (the main structure of artificial neural networks) in the Transformer architecture enabled Transformer models to store abstract facts (such as relationships between inputs) from the training data (Geva et al. 2021). The ability to store abstract facts was another improvement over RNN models wherein the quality and quantity of information remembered by the model were improved.

Transformer models have improved text generation performance compared to previous ML architectures and their performance is typically measured on metrics centered around prediction accuracy. When applying ML to real-world tasks, it is important to design the system for impacted stakeholders. Furthermore, values-oriented design frameworks aim to design systems by integrating the values of stakeholders. For CRTs, such frameworks guided the integration of

students' values into recommendations and provided approach for evaluating CRTs from the perceptions of students.

2.4. Values-Oriented Design Approaches

Various design approaches such as human-computer interaction, value-sensitive design, and value-based engineering focus on the human side of developing and evaluating technology. Of the three design methodologies discussed in this section, a subset of human-computer interaction, human-centered algorithm design (HCAD) (Baumer 2017), is the best framing of the overall approach for my design of the research presented in my thesis. However, HCAD does not address values in detail beyond the presence of values in speculative design, so I also explored value-sensitive design and value-based engineering for inspiration on methods to integrate students' values into the design of my CRT.

HCAD originated from human-centered design (HCD). HCD has a variety of definitions that can be summarized as placing human beings at the center of design such as understanding people holistically through multi-disciplinary collaboration; participatory design i.e. including stakeholders in designing and evaluating; and designing things that are useful, useable, and desirable (Zhang 2009). Zhang's (2009) review of human-centered design approaches discussed the design shift towards human-centeredness since the 1990s. This shift aimed to move away from viewing people as buyers of products and towards viewing people as impacted stakeholders. HCD has since been applied to algorithm design (HCAD), higher education tool development, and product design, all with the goals of creating designs that are built for the

benefit of the people who are, or may be impacted (Baumer 2017; McConvey et al. 2023; Zhang 2009).

Human-centered algorithm design (HCAD) bridges human-computer interaction (HCI) and human-centered design and is applied to the technical aspects of computer algorithms (Baumer 2017; Hartson 1998). After algorithms are developed, evaluations typically focus on analytic performance metrics such as accuracy and speed. This approach is acceptable in theory, but often falls short by not evaluating for potential adverse effects on society (Baumer 2017). The fundamental value of the usability of an algorithm is left out when approaching algorithms strictly to optimize analytic performance (Hartson 1998). As Hartson (1998) highlights, usability varies between people and the diversity of what makes a design useable is a key part to research in the field of HCI.

To improve the usability of algorithms, HCAD incorporates human and social interpretations into algorithm design by taking theoretical, speculative, and participatory approaches (Baumer 2017). Theoretical approaches apply behavioural and social science theories to identify what factors to include when modelling human behaviour. Understanding the results of algorithms designed through theory is also aided by those theories to explain outputs and impacts.

Speculative design approaches are meant to be provocative, explore alternatives to current designs, and may emphasize a particular value or value-tension. Speculative designs are meant to be analyzed for what makes them useful to inform future designs. Participatory design approaches target a short-coming that both the theoretical and speculative approaches are missing – the direct inclusion of people who may be impacted by such designs. By including people in the development and evaluation of algorithms, participatory approaches are meant to aid designers and stakeholders by giving each group a better understanding of the other and the

designed algorithm. When any of the three approaches are used, Baumer (2017) argues that the resulting algorithms should be evaluated using both traditional algorithm performance metrics (accuracy, efficiency) and qualitative results from stakeholders as either on their own is insufficient.

In a review of decision-making algorithms in higher education (McConvey et al. 2023), three pathway advising algorithms used participatory design approaches. Of the three algorithms, “Pathways” (Chen et al. 2022) applied each HCAD design approach. “Pathways” was a tool for generating personalized course pathways for students of a variety of general undergraduate majors and is based on natural language processing (NLP). The programs of each major contained substantial flexibility for electives and contrasts the rigidity of engineering programs. The goal of “Pathways” was to support students’ course planning based on their interests. When students were choosing their courses, Chen et al. (2022) found relevant factors included (in descending order) degree requirements, time schedules, class capacity, practical skills within courses, instructional style, workload, difficulty, and grades. Furthermore, interviews revealed that students often felt overwhelmed from all the factors that affected course selection (Chen et al. 2022). The three HCAD approaches of theoretical, speculative, and participatory were used by Chen et al. (2022) as follows. The CRT was designed with the theoretical support of a constructivist epistemology and social science frameworks for student motivation. The theoretical design approach guided representation of students’ interests to increase students’ motivation. The speculative nature of the tool was that the pathways output by the CRT were designed to be serendipitous, meaning that the pathways were unexpected yet relevant. Participatory design was performed through directly involving students (a primary stakeholder group) in the design and evaluation of the CRT. Results from “Pathways” included aiding student

discoveries of courses interesting to them, and aiding students gain insights into diverse course pathways through visualizing many pathways. The success of the “Pathways” tool demonstrated effective use of all three aspects of HCAD. However, there is room to build upon this work. Since “Pathways” was trained on historical enrollment data, a filter-bubble may arise and result in biases being perpetuated within course pathways. To address the issue of recommendation bias, the use of synthetic training data may be used to control the frequency of training examples and reduce bias. McConvey et al. (2023) concluded from the algorithmic decision-making review that there is much work to be done in HCAD for decision-making algorithms in higher education and highlighted the need for designing from educational theories alongside participatory elements centered around student values.

Value Sensitive Design (VSD) (Friedman 1996; Friedman and Hendry 2019) and Value Based Engineering (VBE) (Spiekermann and Winkler 2020) also recommend participatory design approaches alongside the use of theoretical approaches and stakeholder values. VSD and VBE outlined the need for intrinsic stakeholder values to be integrated into the development of technology. HCAD, VSD and VBE all call for stakeholders to be involved in the development and evaluation of technology through a different approaches.

Value Sensitive Design is a methodology for integrating human values into technologies during the design process (Gerdes and Frandsen 2023). Originally, “values” in VSD were conceptualized as moral and ethical values regarding welfare and justice. VSD’s view of values was later expanded (after a decade) to the current working definition of “what is important to people in their lives, with a focus on ethics and morality” (Friedman and Hendry 2019). These constructs (importance, ethics, and morality) are vast and complex as each person has their own values. Some examples of human values in systems design provided by Friedman and Hendry

(2019) are privacy, universal usability, trust, autonomy, accountability, and environmental sustainability. Within VSD, supporting human values is the design goal.

Although human values are multifaceted, VSD's tripartite methodology consists of conceptual, empirical, and technical investigations to gain an in-depth understanding of values as they pertain to the system being designed. Conceptual investigations are theoretical or philosophical explorations of the central issues surrounding the system. Empirical investigations consist of a variety of mixed methods to understand the human contexts surrounding the system. Technical investigations are meant to explore related technologies for how values are supported or hindered and meant to identify designs that support the values identified by the conceptual investigation. All three of VSD's investigations are needed to "inform and shape and reshape each other" (Friedman and Hendry 2019).

Once the relevant values have been identified, VSD describes methods to design for those values. Seventeen methods for applying VSD are presented in Friedman and Hendry's book (Friedman and Hendry 2019). Of these seventeen methods, *value-oriented semi-structured interviews* and *value-oriented prototypes* (both considered empirical investigations) are the most relevant to my development of a CRT. Semi-structured interviews investigate the values of stakeholders, their views on value tensions, and their concerns surrounding the design.

Prototypes are presented to stakeholders for investigation of value perceptions and the implications of the prototype's potential impacts. I drew inspiration from both of these empirical investigation methods for the evaluation of my CRT since within VSD, values are design goals.

In Gerdes and Frandsen's (2023) thematic literature review of 28 years of VSD research, many themes emerged regarding theoretical contributions to VSD, and an analysis of how VSD is addressed in applied research was presented. As my research aligns with applying VSD concepts,

I will discuss the relevant findings of the applied research. Four types of applied research were identified as “exploration of a design space”, “redesign” excluding prototypes, “redesign” including prototypes and mock-ups, and “technical development and implementation” (Gerdes and Frandsen 2023). Research exploring a new design space aimed to better understand the impact of values on the design of a technology. Both types of “redesigns” were focused on making changes to existing designs to test other features or explore a different design space with the aim to inform future design directions. Technical development and implementation studies focused on the integration of VSD into the process of software development. Of these types of applied research (Gerdes and Frandsen 2023), “redesign” is the best fit to describe my thesis as I redesigned the concept CRT to integrate human values and inform future designs of CRTs based on the results from the developed prototype.

Another methodology to integrate values into designs is Value-based Engineering, which aims to extend non-functional requirements-engineering efforts in corporate settings (Spiekermann and Winkler 2020). VBE builds on top of VSD, machine ethics, HCI, and participatory design to define values as design requirements. Values within VBE are thought of as desires that can represent both positive and negative aspects, where positives are sought out and negatives are avoided. The VBE process consists of identifying the system that will be designed, exploring values associated with the system, engaging stakeholders in the design and iterative feedback of the system, and integrating value requirements with functional requirements (Spiekermann and Winkler 2020). As VBE is meant for a corporate setting (often focusing on business elements); it is not a good fit for research contexts seeking to integrate values into the design of technology.

In summary, HCAD defines 3 design approaches of theoretical, speculative, and participatory design to make algorithms more human-centered through the integration of values but lacks a

definition of values and explicit methods for their integration. Within VSD and VBE, supporting human values is the goal of the design however, VBE situates itself within a corporate setting, which does not align with my research. As I aimed to integrate many human values into the design of my CRT, both VSD and VBE could support my research, however, neither framework has methods specifically meant for CRT or ML applications. As such, with my aim to integrate expectancy-value theory into a CRT, I chose to use HCAD's theoretical and participatory design approaches to integrate expectancy-value theory and course choice motivations, and to engage students in the design and evaluation of my CRT. I drew inspiration from VSD's methods of value-oriented semi-structured interviews and value-oriented prototypes to enhance the participatory evaluation of my CRT. My CRT was thought of as being a value-oriented prototype, which could be used to better understand how students' value affect their course choices. In the next section I review the EVT value framework for my thesis to better understand its representation of values and how prior research has operationalized such values to explore engineering students' motivations.

2.5. Value Framework: Eccles' Expectancy-Value Theory of Achievement Motivation

Eccles' expectancy-value theory (EVT) of achievement motivation (Wigfield and Eccles 2000) is a task value theory for understanding what motivates people to complete a task. EVT explains a person's motivation to perform a task as a combination of the person's expectations about the task and trade-offs between their subjective values about the task. EVT separates subjective task values into attainment, interest, utility and cost. Value representative questions for each EVT subjective task value are listed in Table 1.

Table 1. EVT values and representative questions

EVT Value	Value Representative Question
Attainment	Is the task important to who I am?
Interest	Is the task enjoyable?
Utility	Is the task useful for my future goals?
Cost	How much am I giving up doing the task?

Attainment is how important the task is to a person's identity. Interest is the person's intrinsic curiosity or enjoyment of the task. Utility is the future usefulness of the task to the person. Cost is what the person gives up doing the task. I applied EVT to engineering students' course choice motivation that describes the motivation to choose a course as a combination of expectancy beliefs and subjective task values, displayed in Figure 3.

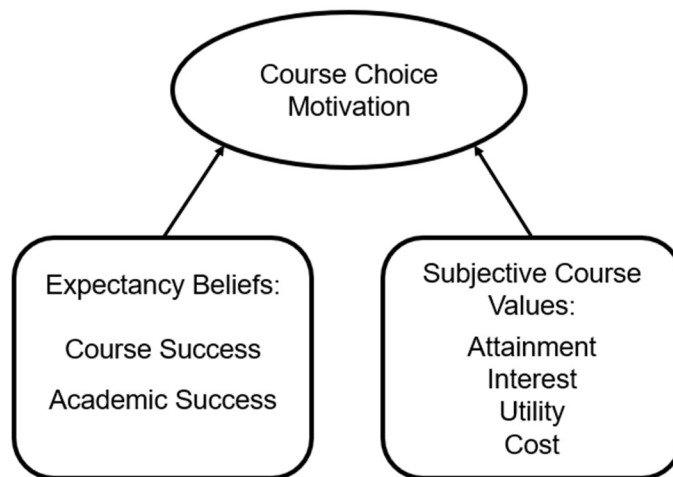


Figure 3. EVT applied to course choice motivation

EVT is aligned with students' balance-oriented approaches to course choice (Ma et al. 2021). The alignment is largely due to the trade-offs that occur as part of the decision-making process to balance a semester of courses and the trade-offs between subjective task values to describe

motivation. Aligning these ideas by their use of trade-offs provides a means to integrate students' motivational values into my CRT. By applying EVT as a value framework for my CRT, I followed HCAD's theoretical design approach, which is also adjacent to the conceptual design investigations described by VSD.

EVT's value constructs of interest, utility, and cost align well with the balance-oriented course choices influences of students' interests, perceived course usefulness (utility), and perceived course difficulty (cost). Ma et al.'s (2021) questionnaire did not ask identity (attainment) related questions. For my research, courses are tasks that students actively choose each semester based on their course choice motivations. As my research is situated within engineering education an additional benefit of using EVT is the prevalence of applied EVT research in engineering education to describe engineering students' motivations for a variety of academic decisions. As such, engineering education literature informed my understanding of EVT in the context of engineering education and guided my operationalization of each EVT value to model engineering students' course choice decisions for my CRT.

2.5.1. EVT in Engineering Education Research

Previous engineering education research has used EVT to investigate engineering student decision making (Matusovich et al. 2010; Ortega-Alvarez et al. 2016; Wu et al. 2020), engineering student persistence (Bego et al. 2022; Matusovich et al. 2010; Robinson et al. 2023; Thomas et al. 2023), engineering student engagement (Lam and Wilson 2024), and to understand engineering student perceptions of assessments (Xu et al. 2025b). By reviewing the findings of

EVT research regarding engineering students' motivations, CRTs can be designed with these in mind to facilitate aspects such as retention and persistence.

Matusovich et al. (2023) conducted a longitudinal investigation of 11 engineering students over 4 years of study to better understand what motivates engineering students to persist in their program. Each participant was interviewed yearly and rated as having high, moderate, or low value for each of Eccles' 4 values. The reasoning behind why participants persisted was always linked to attainment. Although reasons were stated as not always being associated with becoming an engineer, they were always engineering related. Cost was observed to be low for those with high attainment but varied for those with low attainment. This suggests that achieving high attainment may reduce the perceptions of the costs related to studying engineering. The authors concluded, of Eccles' four values, attainment was found to play the most prominent role for persistence in engineering and that the sense of self associated with attainment did not have to be primarily focused on becoming an engineer (Matusovich et al. 2010). CRT research focused on retention could benefit from aiming to discover what courses facilitate attainment and developing a CRT to recommend those courses to engineering students.

Honken et al. (2016) conducted a study of first-year retention of engineering students. The authors found that the top three factors motivating the choice of studying engineering were interest in engineering, job availability, and good pay. Of these three, interest in engineering was a reason some students dropped out of their engineering program. When participant retention was compared based on GPA and level of interest, students with below average GPA had approximately twice the increase in retention compared to above average GPA participants when both had high levels of interest (Honken et al. 2016). In a follow-up study (Robinson et al. 2023), situated (single instance) interest and maintained (many instances) interest were investigated

through a first-year engineering design course with the goal of identifying which aspects of the design course contributed to higher maintained interest, which is a better predictor for long-term student retention. Most participants surveyed had a positive change in the strength of their interest in the engineering profession from the course, which was beyond the situated interest of course topics and activities. Based on the interest-related results from both studies (Honken et al. 2016; Robinson et al. 2023), CRTs designed for the retention of engineering students should facilitate maintained interest in course recommendations.

A study on engineering students' choice of major consisted of interviews and an analysis of the motivations behind their decisions (Ortega-Alvarez et al. 2016). The authors found that the top three coded categories for participant motivations were (in descending order) interest, profession related (attainment), and practicality (utility). Interest had nearly double the quantity of participants compared to professional and practicality, which were nearly the same. This study highlighted how the intrinsic motivation of interest is a common factor for motivating engineering students to choose their major. As interest motivated engineering students to choose their engineering major, it is likely that interest motivates students to choose their concentration within that major as well. This means that a CRT could learn to differentiate interests based on which concentration they are associated with.

When investigating engineering students' motivations for enrolling in more engineering courses, a survey was used to collect student ratings of statements associated with EVT (Wu et al. 2020). Regression models were used to analyse the effects of value groupings on achievement related behaviours. Engineering students were found to have higher persistence than their peers if they had higher confidence in their academic abilities (expectancy), felt success in engineering to be important (attainment), found enjoyment (interest), and felt engineering was useful to their future

goals (utility). The attainment and interest values were best at predicting academic effort of engineering students while interest and utility best predicted planned persistence in the field of engineering after graduation. The authors also highlight that self-efficacy did not significantly impact achievement-related behaviours. Although this study is the closest example of EVT being used to investigate enrollment decisions by engineering students, it did not capture students' motivations for enrolling in specific courses. Wu et al.'s (2020) study did not separate course choice behaviours of engineering students into clusters such as Ma et al.'s (2021) course choice orientations. As such, my research can build upon Wu et al.'s (2020) and Ma et al.'s (2021) work by investigating engineering students' perceptions of EVT values for each course in the computer engineering curriculum, and lay the groundwork for research on course recommendation tools explicitly built with and for engineering students and grounded in EVT.

From these studies it is clear that EVT is commonly used in engineering education research to analyze and understand engineering students' motivations behind their academic decisions; making it a good choice for representing engineering students' course choice values.

Furthermore, these studies highlighted how EVT values vary in importance depending on context and that not all students are motivated by the same values even when making the same kind of decisions. As such, a CRT designed to support engineering students' course choices should model the variations in the importance of values such that each student receives useable recommendations and recommend value-maintained course pathways.

Variations in value importance can be encapsulated within the design of value trade-offs through EVT and a balance-oriented approach to course selection since the idea of what makes a value-balanced semester of courses inherently varies from student to student. This approach of using trade-offs facilitates divergent and convergent recommendations such that different value

variations lead to different pathways while supporting cases of equifinality where different value variations leading to the same (or similar) pathways. In addition, if engineering students are able to determine which course values are most important to them, a CRT can provide recommendations personalized by their value trade-offs to recommend value-maintained course pathways. These premises are the backbones how I modelled engineering students' course choice motivations to develop my values-oriented CRT.

2.5.2. Integrating EVT into Course Recommendations

Engineering students' values are the core aspects that guided course recommendations within my CRT. As my approach to using EVT to model engineering students' course choice motivations is novel, a method was piloted to quantitatively represent utility, cost, and interest within a ML dataset for course pathway recommendations. My approach was evaluated through the usefulness of a CRT trained on the ML dataset. To integrate EVT values into my CRT, I quantified and mapped utility, cost, and interest onto each course within the computer engineering program. I integrated EVT expectancy onto my CRT as expectations of academic success regarding semester courseload and graduation expectations.

The utility and cost associated with each course in the program were quantified through the results of a survey of students' value perceptions of each course within the program. This survey was based on the Expectancy-Value-Cost Light Scale (Schoeffel et al. 2021). The interest value was quantified for each course based on what topics of interest were present in the academic calendar description and course syllabus where available. The topics of interest were obtained from Engineers Canada's list and descriptions of what curriculum topics are required and

optional within Canadian undergraduate computer engineering programs (Engineers Canada 2025a, 2025b). I represented expectancy through the expectations of semester courseload (number of courses in each semester), and time in the program. These expectancy numbers were incorporated into the dataset as well as the logic of the tool to support these expectations when possible. By quantifying utility, cost, interest, and expectancy, I was able to analyze the representation of each construct within course pathways; and further statistically analyze a sample of course pathways.

A key aspect of the representation of values within the training dataset for my CRT is the conceptual blending of maintained interest (Hidi and Renninger 2006, 2026) and EVT. Maintained interest was a better predictor than situational interest in student retention (Hidi and Renninger 2006, 2026). As such, I analyzed simulated course pathways for statistical representations of how maintained each aspect of EVT values was throughout the pathways and labelled each pathway as being “more”, “average”, or “less” maintained. I then combined these categories with the balance-oriented course choice approach (Ma et al. 2021), which described how students make value-related trade-offs during course selection. As such, my CRT was designed so computer engineering students select how important each value aspect is to them, and then the CRT recommends one course pathway (of many possible pathways) that maintains their important value aspects throughout their remaining pathway.

The next chapters sequentially present the methods and results of each phase of this research since the results from each phase informed the methods of the next. First, Chapter 3 presents the mapping of EVT’s utility, cost, and interest values onto courses in the computer engineering curriculum. Second, Chapter 4 uses the value mapping for the construction of a values-oriented dataset of course pathways. Third, Chapter 5 uses the value-oriented dataset to inform the design

and training of the underlying generative machine learning model. Fourth, Chapter 6 uses the developed generative model to holistically evaluate the course recommendation tool through mixed methods.

Chapter 3: Value Mapping

This chapter presents the methodology and results for the first phase of my study. In this phase, I mapped EVT values onto each course within the University of Manitoba computer engineering program, highlighted in Figure 4.

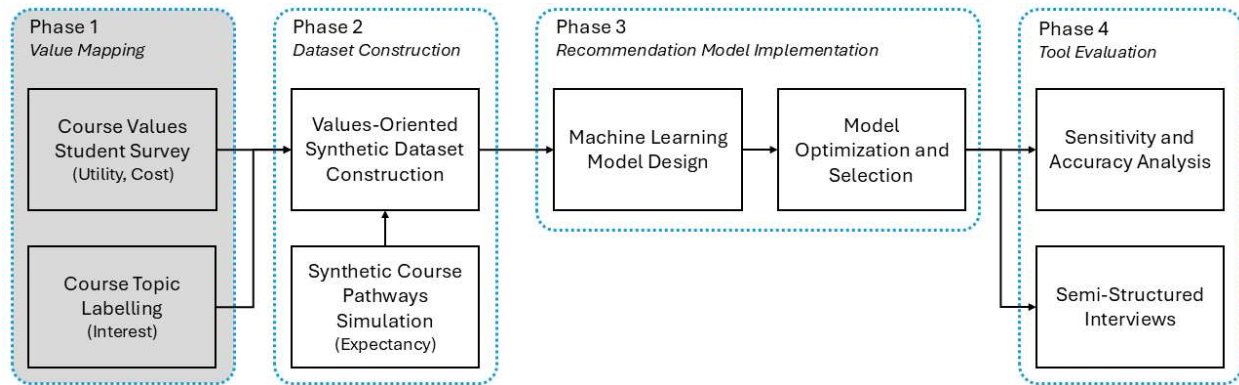


Figure 4. Phase 1: Value Mapping

The purpose of this value mapping was to create the basis of the values-oriented information used to train the machine learning model within the course recommendation tool. I used two approaches to gather EVT data. A survey of students' value perceptions of courses (in terms of utility and cost) and a content analysis of syllabi and academic calendar descriptions of each course for curriculum topics (interest). In this chapter, I first discuss the survey design followed by my survey validation that focuses on the utility and cost values of each course. Then I present my process for the content analysis, which identified courses' topics of interest based on subjects for curriculum accreditation. Next, I present the results of the survey and discuss how my findings could inform preliminary-year students to enroll while balancing the value of utility of

cost. The end of this section reports my results from the interest topic analysis and a brief discussion of how the combined results from the survey and content analysis will be used to provide the basis of values-oriented learnable information for training my machine learning model.

3.1. Survey of Students' Course Values Perceptions

To understand students' perceptions of utility and cost values associated with specific courses, I surveyed computer engineering students at the University of Manitoba. My survey was based on the Expectancy-Value-Cost (EVC) Light scale (Schoeffel et al. 2021), which was developed to measure student motivation in computing courses according to Eccles' expectancy-value theory. Schoeffel (2021) suggested that results from the EVC Light scale could be used as inputs to machine learning algorithms to make predictions of student behaviour, which further aligns the survey with my CRT. The EVC Light scale is a simplified scale, combining similar items from the original EVC scale (Kosovich et al. 2015). Previous research using either the EVC or EVC Light scale tended to focus on a single course or a small cluster of related courses. Neither scale has yet been used for a curriculum-wide investigation of student motivation. As such, my thesis is one of first research projects to apply this motivation-mapping scale onto a curriculum. I followed the EVC survey user guide (Barron et al. 2017) to design the survey with Likert-scale items. I used a 6-point Likert-scale from strongly disagree to strongly agree, without a neutral option, for the survey items. A benefit of using the Likert-scale is the ease of translating the discrete responses to numerical representations for integration into the machine learning dataset. The survey items for the constructs of utility and cost from the EVC Light scale were used alongside two additional statements I added to the survey (marked by * in Table 2).

Table 2. Value perceptions survey statements

Value	Survey Statement
Utility1	I think that the course lessons are useful for the program.
Utility2	I understand the importance of this course in the program.
Utility3*	I think that the course lessons are useful for my career.
Cost1	The course required a lot of time and I had to give up doing other activities.
Cost2	I did not get time to dedicate myself to this course.
Cost3*	I think that the course lessons are too difficult.

I added one extra utility item and one extra cost item to further study aspects of each construct (utility and cost) that I felt were not represented by the EVC and EVC Light scales. The additional utility item focuses on the perceived usefulness of the course for the participant's career while the additional cost item focuses on the perceived difficulty of the course. As this survey of students' value perceptions of courses was framed as a retrospective reflection, the EVC Light construct of expectancy was not used because each item focused on future expectation. Validation of these additions to the survey scale is discussed below in Section 3.2.

Survey participants were current computer engineering students at the University of Manitoba who had completed at least 60 credit hours of courses (approximately 2 years) as of the Fall 2025 semester. These inclusion criteria allowed for a significant number of eligible participants with enough experience of courses within the program to provide responses about many courses.

The survey included 41 core courses (including equivalent course options), and 3 types of electives (Technical, Natural Science, and Complementary Studies). Each elective type was further split into 2 sub-categories, to create differentiating representations of elective selection. Technical electives were split into being offered by either the department of Electrical and Computer Engineering or the department of Computer Science. Natural science electives were split into those offered by the departments of Chemistry or Physics, which are departments with

core courses in the program, and those offered by another department. Complementary studies electives are courses from either the Faculty of Arts or the Faculty of Management and were divided as such. Choosing this approach to represent electives as groupings directly affects the specificity of the machine learning model by generalizing electives: in other words, the specificity could have been improved by including individual elective courses rather than these categories. However, the category approach is favorable because of the expected low sample size of any individual elective. Given the number of survey participants, the results for any given individual elective course would not have yielded a sufficient sample size to adequately represent the value perceptions of each course. Validation of my survey with attention to my additional items is discussed in the following section and follows the validation methods used to validate the EVC Light scale.

3.2. Survey Validation

As my additional survey items extend the EVC Light scale (Schoeffel et al. 2021), the results of the modified survey scale needed to be validated. I applied the same validation process used for validating the EVC Light scale (Schoeffel et al. 2021) with respect to the original EVC scale (Kosovich et al. 2015). The validation examined whether my additional survey items measured the utility and cost constructs respectively. Schoeffel et al. (2021) measured the internal consistency of each construct with McDonald's Omega, convergent and discriminant validity through the intercorrelation of survey items, and performed confirmatory factor analysis (CFA) through comparative fit index (CFI), root mean square error of approximation (RMSEA), standardized root mean square residual (SRMR), and factor loading. Responses for all courses were combined for this validation.

My survey displayed a strong internal consistency with an Omega of 0.942 for the construct of utility and a good internal consistency with an Omega of 0.851 for the construct of cost. Within CFA, the model has an excellent fit with a CFI of 0.986, which is comparable to EVC Light's CFI of 0.988. The error of the model is moderate with an RMSEA of 0.088 although this may be due to my survey validation encompassing each course in the curriculum (a broad range of content) compared to EVC Light that focused on 8 closely related courses (similar content). Conversely, the SRMR of my survey (that measures the difference between observed and predicted correlations) is 0.0, which is ideal. Although ideal, the small number of participants may have simplified the statistical modelling as each of the 14 participants contributed to the 423 individual course responses used for validation. When investigating the correlations between items (1 is strongly positively correlated, 0 is not correlated, -1 is strongly negatively correlated), displayed in Table 3, it was clear that both additional statements are correlated with their related survey items.

Table 3. Survey item intercorrelation

	Utility1	Utility2	Utility3*	Cost1	Cost2	Cost3*
Utility1	-					
Utility2	0.893	-				
Utility3*	0.834	0.804	-			
Cost1	-0.169	-0.110	-0.166	-		
Cost2	-0.249	-0.177	-0.260	0.601	-	
Cost3*	-0.191	-0.163	-0.267	0.583	0.757	-

The Utility3* item (the first new statement added for my modified survey scale) had a high positive correlation with Utility1 and Utility2, and had weak negative correlations to the Cost items. The Cost3* item (the other new statement added) showed a moderately positive

correlation to Cost1, however it was similar to the correlation between Cost1 and Cost2. The correlation between Cost3* and Cost 2 was highly positive whereas the correlations to the Utility items were weakly negative. After viewing the intercorrelation of items, I investigated the item factor loading to see how each item is correlated to the two latent factors within the survey (the constructs of utility and cost), displayed in Table 4.

Table 4. Survey item factor loading

Item	Factor 1	Factor 2
Utility1	0.953	-0.112
Utility2	0.931	-0.050
Utility3*	0.855	-0.165
Cost1	-0.083	0.675
Cost2	-0.147	0.871
Cost3*	-0.124	0.849

All Utility items are highly positively correlated to Factor 1 and weakly negatively correlated to Factor 2. Cost2 and Cost3* are strongly positively correlated to Factor 2 whereas Cost1 has a moderate positive correlation to Factor 2. Each cost item is weakly negatively correlated to Factor 1. From these results it is clear that Factor 1 is utility while Factor 2 is cost.

Based on the survey validation results, my additional items of Utility3* and Cost3* were consistent and reliable for measuring utility and cost respectively. Based on the intercorrelations of items and the item factor loading, Cost3* better integrates with EVC Light than Utility3* through stronger positive correlations.

3.3. Survey Results

The survey results provided values-focused course information to aid in the construction of a course recommendation dataset for ML. Additionally, I used the results to analyze UM's Computer Engineering program. I analyzed the program by combining the survey items for utility and cost respectively, plotting each course on utility and cost axes, and providing examples of how the survey data could be used to recommend courses to preliminary-year engineering students.

The survey of computer engineering students' value perceptions of their completed courses had 14 participants. The number of responses received for a specific course differed between courses since participants were at different stages of the computer engineering program. Figure 5 shows the number of student responses received for different courses, with the responses per course ranging from 0 to 14.

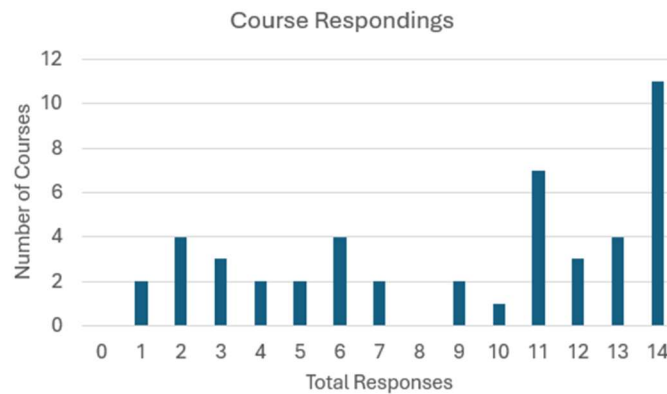


Figure 5. How many courses had a number of responses

No course had 0 or 8 responses, and 11 of 47 courses had responses from all 14 participants.

Figure 6 presents the decremental distribution of course responses to visualize the percentage of courses that would be left out of analysis if a minimum number of responses was set (i.e. if the minimum required number of responses was set at just 1, then 100% of courses would be included in the analysis).

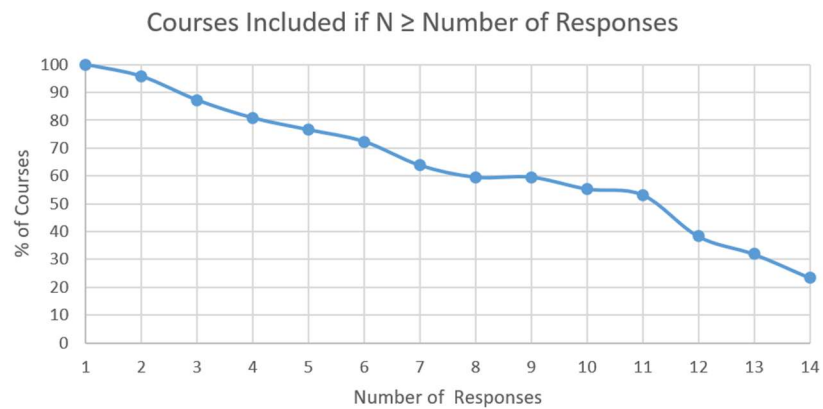


Figure 6. Percent of courses included when $N \geq$ the number of responses

Further in this Section, I set $N \geq 5$, which included 77% of courses. To quantify courses' utility and cost values, I used the average of all responses received for each respective course, where each response was quantified using the scale in Table 5.

Table 5. Survey quantization scale

Rating	Strongly Disagree	Disagree	Slightly Disagree	Slightly Agree	Agree	Strongly Agree
Number	0.0	0.2	0.4	0.6	0.8	1.0

The mean, standard deviation, minimum, and maximum quantified value representations of all courses in the computer engineering curriculum at the University of Manitoba are shown in

Table 6.

Table 6. Statistics of course utility and cost representations

Value Parameter	Mean	Standard Deviation	Minimum	Maximum
Utility1	0.731	0.178	0.311	1.0
Utility2	0.752	0.177	0.311	1.0
Utility3*	0.681	0.191	0.233	1.0
Cost1	0.517	0.154	0.240	0.9
Cost2	0.392	0.133	0.169	0.8
Cost3*	0.378	0.167	0.156	0.9

The utility of courses in the program tend to be near the level of “Agree” meaning that the courses in the program are perceived to have utility. The cost value results tended to align with “Slightly Disagree”, meaning that courses tended to not have much perceived cost. Considering that the step size between ratings in the scale is 0.2, the standard deviations of each value suggest that participants are close in agreement. It is important to note that the items with the highest standard deviation within utility and cost respectively, the third items, were ones I added to the survey. This, in turn, suggested that the additional statements might be providing more distinct learnable information for the machine learning model, compared to the EVC Light items. The minimum and maximum representations show considerable range within the curriculum with 0.156 being the lowest minimum and 1.0 being the highest maximum. These findings indicate that there is significant room for value representations to vary within each semester based on which courses are taken.

As one of my goals for the survey was to investigate the utility and cost representations of the whole computer engineering curriculum, I needed to combine the survey items into their value constructs, which requires construct internal consistency. To measure the internal consistency of the survey constructs for each course, Cronbach's alpha (Cronbach 1951) was computed using responses from the six survey items. To be able to combine the results from the items within each value construct (i.e. valid internal consistency) Cronbach's alpha should, at minimum, be greater than or equal to 0.5 as anything below is generally considered unacceptable (Taber 2018; Tavakol and Dennick 2011).

Using $\alpha \geq 0.5$ as the target, I started increasing the minimum number of responses required for each course to be included in the value analysis of the curriculum as courses with small samples are the most likely to have weak internal consistency and could change if new data was added.

Table 7 displays the number of courses that have a minimum number of responses and fall under corresponding bins for alpha for the utility value and Table 8 similarly displays this information with respect to cost.

Table 7. Cronbach's alpha of utility for courses with at least N responses

N	$\alpha \geq 0.5$	$\alpha < 0.5$
1	42	5
2	42	3
3	39	2
4	37	1
5	36	0

Table 8. Cronbach's alpha for cost for courses with at least N responses

N	$\alpha \geq 0.5$	$\alpha < 0.5$
1	45	2
2	45	0
3	41	0
4	38	0
5	36	0

For utility, at least 5 responses were needed to only include courses considered to have an acceptable alpha whereas for cost only 2 responses are needed. Using maximum of the two options results in $N \geq 5$, which accounts for 77% of courses in the survey. Details on the distribution of courses encompassed by $\alpha \geq 0.5$ can be found in Appendix C. Going forward, only courses with at least five responses were used for analysis to uphold acceptable internal consistency for the utility and cost values.

To compare the combined quantized value representations of each course in the program, courses were ranked by the sum of their utility, sum of their cost, and sum of their utility minus the sum of their cost. By including the difference between utility and cost, I present insights on what a value trade-off might look like compared to each value on their own. However, viewing either utility or cost independently can also be thought of as a value trade-off (trading off one value completely for the other). As such, utility minus cost demonstrates an example of how courses are perceived when more than one value is considered (as many CRTs only focus on one concept at a time and my CRT aims to represent many). For simplicity, the four highest and lowest ranking courses based on their combined utility, cost, and utility minus cost representations are listed in descending order in Table 9.

Table 9. Descending value rankings excluding electives, $N \geq 5$

Utility	Cost	Utility – Cost
ECE 2400 Engineering Algorithms 1	MATH 2136 Mathematics for Computer Engineers	COMP 1012 Computer Programming for Scientists and Engineers
COMP 1012 Computer Programming for Scientists and Engineers	ECE 3780 Signal Processing 1	ECE 4240 Microprocessing Interfacing
ECE 4240 Microprocessing Interfacing	MATH 2132 Engineering Mathematical Analysis 2	ECE 3610 Microprocessing Systems
ENG 1450 Introduction to Electrical and Computer Engineering	MATH 2130 Engineering Mathematical Analysis 1	MATH 1510 / MATH 1500 Calculus 1
...	...	...
ENG 1460 Introduction to Thermal Sciences	STAT 2220 Statistics for Engineers	CHEM 1100 Introduction to University Chemistry 1
ENG 1440 Introduction to Statics	ENG 3000 Engineering Economics	ENG 1460 Introduction to Thermal Sciences
CHEM 1100 Introduction to University Chemistry 1	COMP 1012 Computer Programming for Scientists and Engineers	ENG 1440 Introduction to Statics
CHEM 1122 Introduction to Chemical Techniques for Engineering 1	ECE 4240 Microprocessing Interfacing	CHEM 1122 Introduction to Chemical Techniques for Engineering 1

These results do not include elective courses, and each course had 5 or more responses ($N \geq 5$).

The highest utility courses appear to focus on the interests of computer programming and digital systems design, although appearing at varying levels throughout the program (1000, 2000, and 4000). The lowest utility courses were preliminary year courses that do not have topics of interest that align with the computer engineering curriculum and do not serve as prerequisites within the program, so this result is expected. The highest cost courses were math heavy courses,

including all 3 mathematics courses above the 1000 level and signal processing, a course the 3 mathematics courses are prerequisites for. The lowest cost courses are offered by a variety of departments inside and outside of the Faculty of Engineering, and at all 4 levels, as such no pattern was found. A trade-off of utility minus cost was investigated with two of the highest being upper-level courses focused on the topic of computer architecture while the lowest courses were preliminary year courses, which are not prerequisites for other courses in the program. These three ranking examples highlight the variations in value perceptions of computer engineering students under different value trade-offs and begin to layout the idea of balancing course values within a semester by pairing high and low value courses. As the previous table excluded electives, Table 10 presents the course rankings of the elective courses following the same value trade-offs.

Table 10. Descending value rankings of grouped electives, $N \geq 5$

Utility	Cost	Utility – Cost
Technical: ECE	Technical: ECE	Technical: ECE
Technical: COMP	Science: Other	Technical: COMP
Science: CHEM or PHYS	Technical: COMP	Complementary: ARTS
Science: Other	Science: CHEM or PHYS	Science: CHEM or PHYS
Complementary: ARTS	Complementary: ARTS	Science: Other

For my research, electives were grouped into the 3 types of required electives required by the computer engineering program and each divided into two distinct sub-types. The sub-type of complementary studies electives for courses in the Faculty of Management are not included in the results as less than 5 responses were received, meaning this category was excluded to maintain internal construct consistency. Technical electives had the highest perceived utility,

followed by natural science, and finally complementary studies. Electrical and Computer Engineering technical electives had the highest perceived cost while complementary studies had the lowest cost. However, due to the high perceived utility of technical electives, the utility minus cost trade-off was highest for technical electives, followed by complementary studies, and then natural science. These results can potentially help inform students' choices of elective courses based on these examples of value trade-offs.

After ranking courses based on value trade-off examples, the next analysis I performed consisted of plotting each course in axes of utility and cost. I separated core courses by recommended enrollment year based on the program's recommended course flowchart and plotted elective courses separately from the yearly plots. As this is a summative combination of utility and cost each into one number, 0 still represents Strongly Disagree while 3 represents Strongly Agree and the step size is now 0.6. Figure 7 displays 6 plots, that highlight the yearly core courses and electives (5 plots), and as a combination of all results (1 plot).

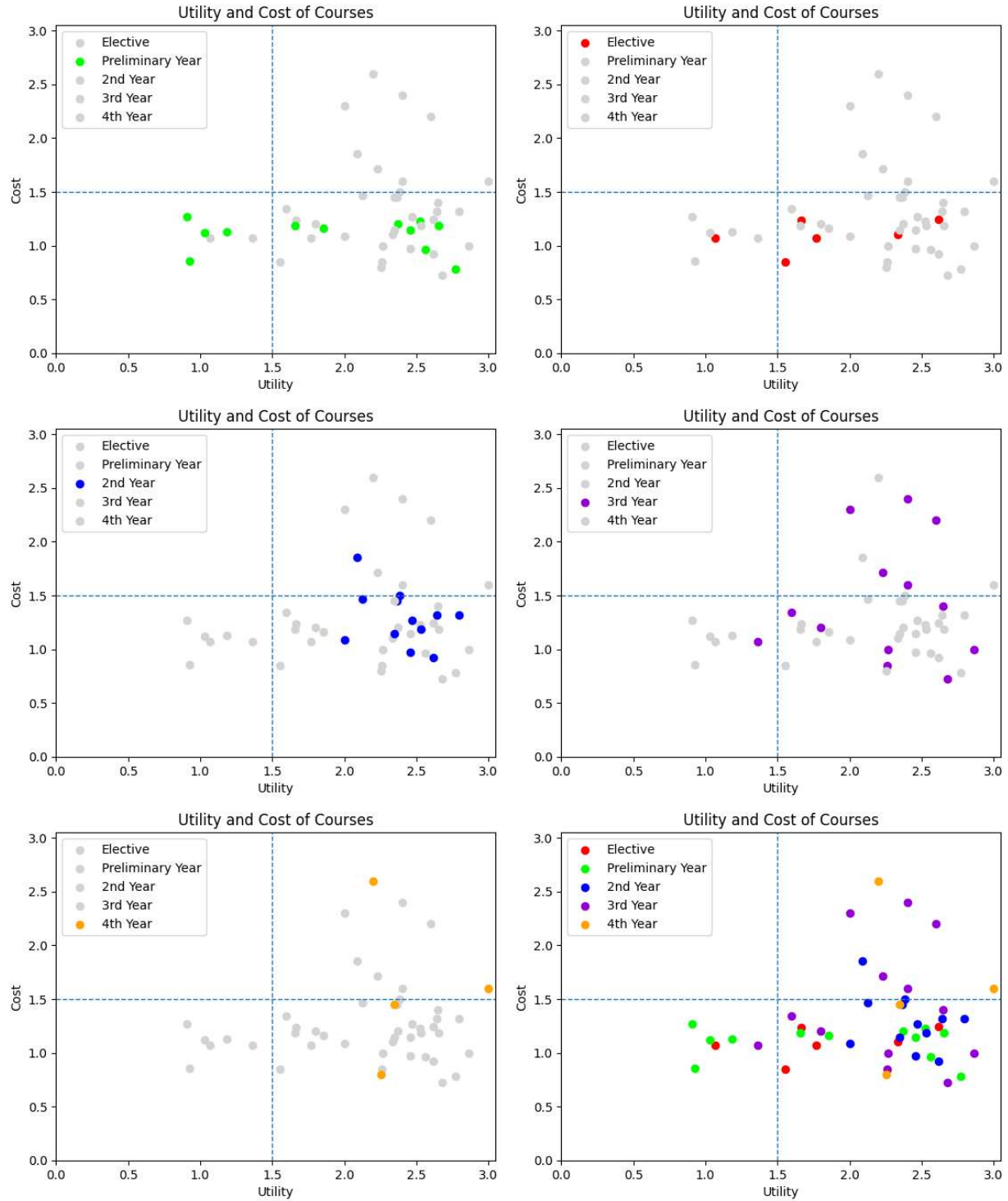


Figure 7. Combined utility and cost for each course by year and electives

Preliminary year and elective courses have similar ranges for both utility and costs where utility spans from around 1.0 to 2.7 while cost has a narrower range from approximately 0.7 to 1.3.

Second year courses tend to be clustered more tightly in comparison ranging from 2.0 to 2.7 on utility and 0.9 to 1.9 on the cost axis. Third year courses are quite spread out and courses with higher cost begin to appear with ranges of 1.4 to 2.9 and 0.6 to 2.5 on the utility and cost axes respectively. Finally, the fourth-year courses, which did not have many responses and only represent four courses, are sparsely spread out from 2.2 to 3.0 utility and 0.7 to 2.7 cost.

Observing the program as a whole, most perceptions of courses were clustered near “Slightly Disagree” for cost and “Agree” for utility.

3.3.1. Examples of Maintaining Course Values in the Preliminary Year

The survey findings present the range of values across the computer engineering program and varying sub-ranges of core courses by year and of electives. As this section focuses on the preliminary year, maintaining values is similar to finding a balance between the amount of each value in the Fall and Winter semesters. In the preliminary year there is more space for value imbalances in utility than in cost as the range in Figure 7 is wide for utility and narrow for cost. For elective courses that have a small range for cost, courses could be strategically used to balance the utility of a semester. The second-year courses were somewhat clustered together in both utility and cost, which make them the most stable year of courses. This stability may be beneficial as some preliminary-year and third-year courses may be taken with second-year courses. Third year courses have a large span and may likely overshadow the focus of balancing when taking third year courses in the same semesters. The introduction of higher cost courses in

third year may require more attention to balancing to avoid a high-cost semester, which may overwhelm students.

Overall, the survey results show that the majority of courses reside in the “Agree” half of utility or in the “Disagree” half of cost. In general, this suggests that courses within the computer engineering program are perceived as useful (utility) and somewhat without cost. These findings also demonstrate that balancing utility and cost (as one-dimensional constructs) changes between years with preliminary year being easier to balance cost and harder to balance utility, while second would be relatively easier to balance both, and third year becomes harder to balance both values due to the size of the course value ranges.

The differences between sub-ranges of each grouping of courses highlights the complexity of seeking maintained value trade-offs throughout each semester. For training a machine learning model, the different sub-ranges between groupings need to be learned to generate maintained value trade-offs across semesters. However, in the case of my CRT, each survey item is treated independently, further adding complexity through a six-dimensional space of the three utility and three cost value items (integrating the interest value further increases the dimensionality).

Focusing specifically on the preliminary-year courses can lead to informative examples of value-balanced course choices. Before students at the University of Manitoba are admitted to an engineering program, ten of the twelve preliminary year courses must be completed. The recommended course flowchart in Appendix A suggests taking seven courses in the fall semester and five in the winter semester. Alternatively, taking six courses in each semester would also complete the twelve courses in a year. However, as only ten are needed for program entry, many students opt to take five courses each semester. Table 11 outlines the value rankings for utility,

cost, and the utility minus cost trade-off. All of the preliminary year courses had at least 9 survey responses.

Table 11. Descending value representations of preliminary-year courses, $N \geq 9$

Rank	Utility	Cost	Utility – Cost
1	COMP 1012 Computer Programming for Scientists and Engineers	CHEM 1122 Introduction to Chemical Techniques for Engineering 1	COMP 1012 Computer Programming for Scientists and Engineers
2	ENG 1450 Introduction to Electrical and Computer Engineering	MATH 1710 / MATH 1700 Calculus 2	MATH 1510 / MATH 1500 Calculus 1
3	MATH 1510 / MATH 1500 Calculus 1	ENG 1430 Design in Engineering	ENG 1450 Introduction to Electrical and Computer Engineering
4	MATH 1710 / MATH 1700 Calculus 2	ENG 1450 Introduction to Electrical and Computer Engineering	MATH 1210 Classical and Linear Algebra
5	MATH 1210 Classical and Linear Algebra	Written English Requirement for Engineering Students	MATH 1710 / MATH 1700 Calculus 2
6	ENG 1430 Design in Engineering	PHYS 1050 Physics 1: Mechanics	ENG 1430 Design in Engineering
7	PHYS 1050 Physics 1: Mechanics	MATH 1210 Classical and Linear Algebra	PHYS 1050 Physics 1: Mechanics
8	Written English Requirement for Engineering Students	ENG 1460 Introduction to Thermal Sciences	Written English Requirement for Engineering Students
9	ENG 1460 Introduction to Thermal Sciences	ENG 1440 Introduction to Statics	CHEM 1100 Introduction to University Chemistry 1
10	ENG 1440 Introduction to Statics	MATH 1510 / MATH 1500 Calculus 1	ENG 1460 Introduction to Thermal Sciences
11	CHEM 1100 Introduction to University Chemistry 1	CHEM 1100 Introduction to University Chemistry 1	ENG 1440 Introduction to Statics
12	CHEM 1122 Introduction to Chemical Techniques for Engineering 1	COMP 1012 Computer Programming for Scientists and Engineers	CHEM 1122 Introduction to Chemical Techniques for Engineering 1

In the case of taking all twelve courses in the preliminary year, as suggested in the ECE course flowchart, these results could aid in creating a balanced set of two semesters for either trade-off. For simplicity, a student could pair the highest and lowest value courses together while ensuring that Calculus 1 and is taken in the fall semester, so the prerequisites are met to take Calculus 2 in the winter semester (note that PHYS 1050 is a corequisite to Calculus 2). However, as this is not the case for many students, the carry-over of untaken courses into the next year begins to unravel into a complex problem of finding value-maintained course pathways that support students' value trade-offs. This unravelled complexity is where academic advisors and course recommendation tools are of assistance to students.

Sample preliminary-year semesters with approximately maintained value trade-offs are displayed in Table 12, Table 13, and Table 14 with the value ranking of each course. Each of these tables assumed taking 6 courses a semester. This assumption required that Calculus 1 and 2 (MATH 1510 and MATH 1710) were taken in the Fall and Winter semesters respectively and those course value rankings were initially used to balance the value across the two semesters from pairing up courses following the pattern $(1, n)$, $(2, n - 1)$, $(3, n - 2)$, etc.

Table 12. Utility maintained example of preliminary year pairings

Utility			
Fall		Winter	
Rank	Course Code	Rank	Course Code
3	MATH 1510	4	MATH 1710
10	ENG 1440	9	ENG 1460
2	ENG 1450	1	COMP 1012
11	CHEM 1100	12	CHEM 1122
5	MATH 1210	6	ENG 1430
8	Written Req.	7	PHYS 1050

Table 13. Cost maintained example of preliminary year pairings

Cost			
Fall		Winter	
Rank	Course Code	Rank	Course Code
10	MATH 1510	2	MATH 1710
3	ENG 1430	11	CHEM 1100
1	CHEM 1122	4	ENG 1450
12	COMP 1012	9	ENG 1440
5	Written Req.	6	PHYS 1050
8	ENG 1460	7	MATH 1210

Table 14. Utility minus cost maintained preliminary year pairings

Utility – Cost			
Fall		Winter	
Rank	Course Code	Rank	Course Code
2	MATH 1510	5	MATH 1710
11	ENG 1440	8	Written Req.
3	ENG 1450	1	COMP 1012
10	ENG 1460	12	CHEM 1122
6	ENG 1430	4	MATH 1210
7	PHYS 1050	9	CHEM 1100

The value-balanced examples show what maintained value approaches to course selection can look like when only considering the preliminary year and based on the survey results. Although these examples may be helpful to preliminary-year students during their course planning, their engineering program only requires 8 such courses to be completed to be eligible for program entrance. In cases where all 12 courses are not taken, there are two key factors that restrict course planning. The first is that the courses that were taken are prerequisites for specific courses while other courses may require the untaken courses as prerequisites. The second factor is that the courses that were not taken still needed to be completed. Both of these factors affect the sub-ranges for values in their next year. The growth in complexity as more courses are available is exactly why my CRT can be an additional support for engineering students to aid in navigating their course planning while maintaining their value trade-offs throughout each year. As my CRT aimed to maintain utility, cost, and interest, the next section presents how I quantified interest for each course from a directed content analysis.

3.4. Course Topic Labelling

To complement the survey data that led to course representations for utility and cost, and further integrate EVT into my training data, I identified and labelled the presence of topics of interest within core courses taken by computer engineering students at the University of Manitoba based on the content in course syllabi and academic calendar descriptions. This approach aligns with previous research on the development of CRTs, which has used curriculum subjects in a similar manner to represent topics of interest (Pardos et al. 2018). I used course syllabi for all 18 courses from the Department of Electrical and Computer Engineering (ECE), and I used 22 academic calendar descriptions for courses outside of the ECE department.

I coded the syllabi and calendar descriptions following a directed content analysis by using subject categories identified by Engineers Canada (Engineers Canada 2025a, 2025b). The subjects are required for computer engineering program accreditation and are divided into compulsory and optional subjects. Compulsory subjects include mathematics, probability and statistics, computation methods, engineering design process, electronics, digital system design, computer architecture, program design and data structures, operating systems, and software engineering. Optional subjects include but are not limited to statics and dynamics, mechanics, basic electromagnets, advanced mathematics, databases and file systems, computer communication, digital signal processing, artificial intelligence and expert systems, and mechatronic design. These broad subjects outlined by Engineers Canada included a description of which topics are typically included within each category, which I followed during the coding process.

As elective courses broadly vary within each category of elective within the program (Technical, Natural Science, and Complimentary Studies), I chose to represent elective courses by generalizing from required courses offered by the same department. This meant that I represented electives as containing all topics that were present in required courses from the same department. My coding process identified the presence of topics. As such, the number or length of coded sections for a course did not affect how I represented topics of interest in each course.

3.5. Course Topic Labelling Results

The course topic labelling results focus on the number of courses containing each topic.

Following the course topic labelling I present an example of a quantified representation of a

course based on the survey and topic labelling results. Table 15 outlines the 10 most commonly coded subjects with the 4 courses being the minimum number of occurrences and 12 being the maximum.

Table 15. Interest coding results

Engineers Canada Accreditation Subjects	Subject of Interest Codes	Number of Courses Coded
Basic Studies Compulsory	Engineering Design Process	11
	Mathematics	7
Computer Engineering Compulsory	Program Design and Data Structures	12
	Digital Systems Design	6
	Computer Architecture	6
	Electronics	6
	Software Engineering	5
Computer Engineering Optional	Mechatronic Design	12
	Computer Communications	6
	Distributed Systems	4

The three most commonly coded subjects were program design and data structures, mechatronic design, and engineering design process, all of which occurred in 11-12 courses. The remaining subjects tended to occur in roughly 6 courses. As the goal of my CRT is to recommend value-maintained course pathways, the more courses that contain an interest topic, the more combinations are possible. Conversely, as course pathways take a minimum of 8 semesters (4 years), topics with less than 8 courses will likely need to be spread out such that 1 course is taken in each semester to be maintained. Figure 8 is an example of how coding was performed on the syllabus for the course ECE 2400 Engineering Algorithms 1 offered in the Winter 2025 semester.

ECE 2400 – Engineering Algorithms 1

Winter 2025

Course Objectives

The objectives of this course are to introduce students to common engineering algorithms. Students will develop the ability to understand, evaluate, analyze, design, and implement a wide array of generally useful algorithms. These skills will enable students to engineer their own algorithms in the future, or properly use existing engineering algorithms to solve a wide range of problems while knowing what to expect as problem sizes scale.

Course Content

The following topics will be covered:

- Mathematical Foundations
- Asymptotic Complexity Analysis
- Divide-and-Conquer
- Dynamic Programming
- Greedy Algorithms
- Graph Algorithms
- Select Topics

Coding Legend	
	Engineering Design Process
	Mathematics
	Program Design and Data Structures

Figure 8. Coded ECE 2400 syllabus sections with highlighted topics

Three topics were present in the description of the course content, “Engineering Design Process” in red, “Mathematics” in blue, and “Program Design and Data Structures” in green. Bringing together the survey results with the topic labelling, each course has quantitative representations for utility, cost, and interest. Table 16 provides an example of how a course is represented numerically for utility, cost, and interest.

Table 16. Numerical course representation of ECE 2400

Value Factor	Numerical Representation
Utility1 – Useful to the program	0.967
Utility2 – Important to the program	0.943
Utility3 – Useful for a career	0.886
Cost1 – Gave up doing other activities	0.543
Cost2 – Not enough time to dedicate	0.400
Cost3 – Lessons are too difficult	0.371
Interest1 – Engineering Design Process	1
Interest2 – Mathematic	1
Interest3 – Program Design and Data Structures	1
Interest4 – Digital Systems Design	0
Interest5 – Computer Architecture	0
Interest6 – Electronics	0
Interest7 – Software Engineering	0
Interest8 – Mechatronic Design	0
Interest9 – Computer Communications	0
Interest10 – Distributed Systems	0

With the curriculum's courses labelled with which topics of interest value are present and with their quantified representations for the utility and cost values, the collection of program's courses' value data was concluded. At this point, the course values are ready to be integrated into the construction of a training dataset to provide a machine learning model with values-oriented training data that labels course pathways based on the numerical representation of utility, cost, and interest as they change throughout each pathway. The next chapter discusses the decisions that went into transforming the EVT values data from the survey and content analysis into values-oriented course pathways used to train a machine learning model to recommend course pathways with maintained value trade-offs.

Chapter 4: Dataset Construction

In this chapter, I build on my survey and topic labelling results by combining the quantified value representations of utility, cost, and interest with synthetic course pathways into a supervised learning training dataset for values-oriented course recommendation, highlighted in Figure 9.

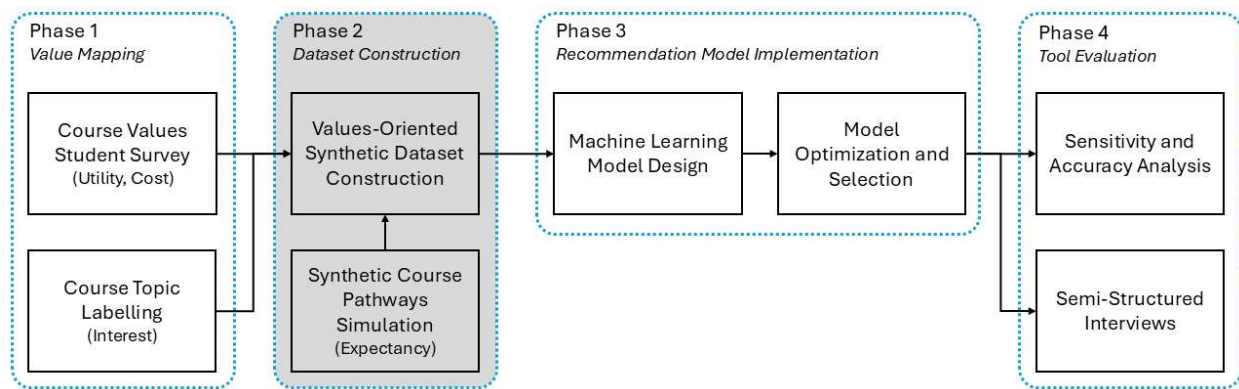


Figure 9. Phase 2: Dataset Construction

To construct the ML dataset, I began with the simulation of synthetic course pathways through UM’s computer engineering program. Then, I labelled each pathway with numerical representations of each value factor based on the survey results for utility and cost, topic labelling results for interest, and enrollment maximums alongside pathway lengths for expectancy. I specifically focused on the representation of maintained values across semesters in the course pathway. As such, I categorized course pathways with similar amounts of maintained value by labelling course pathways “more”, “average”, and “less” maintained value.

4.1. Simulating Synthetic Course Pathways

Synthetic dataset generation has been previously used in engineering education research where datasets for a specific machine learning application do not exist (Singh et al. 2023). In the case of my research, no such dataset of course pathways through the University of Manitoba's computer engineering program exists that was practically and ethically accessible. Additionally, as discussed in Section 2.3, using synthetic data can avoid the filter-bubble biases caused by training on historical enrollment data. As such, I created a synthetic dataset by modelling the curriculum and simulating random course pathways from no completed courses to satisfying all program requirements.

To generate synthetic course pathways, I constructed a directed graph data structure to model the prerequisite structure (Molontay et al. 2020) of the computer engineering program at the University of Manitoba. Graph nodes represented courses and directional graph edges represented a prerequisite relationship (e.g. if CourseA is a prerequisite for CourseB, then $\text{CourseA} \rightarrow \text{CourseB}$). Additional logic was added to handle corequisites (a prerequisite that can be jointly taken instead of only prior), equivalent courses, and semester specific courses.

Pathways were generated through random valid course selections for each semester until all program requirements were met. Initially, these random pathways were unable to complete the program within 4 years, so I added choice logic added to commonly prioritize the final year capstone design course, which was a bottleneck in simulation, and to commonly prioritize the course's prerequisites. As such, the simulated choice logic selected prerequisites for capstone design 90% of the time when available and the prerequisites for those prerequisites 70% of the time when available. This bottleneck was due to the capstone course being 2 semesters long, only beginning in the fall semester, and having many prerequisite courses.

I simulated pathways with the criteria that: all required courses are taken, electives are appropriately satisfied for all 3 subcategories of electives (technical, natural science, and complementary studies). To be considered valid, pathways must have been 7 or fewer years in length, and all semesters included enough courses to be considered a full-time student with the exception that the final two semesters may be part time (in accordance with program policy).

These pathways did not consider failing courses, summer semesters (which typically had minimal course offerings), Advanced Placement or International Baccalaureate transfer credits, or semesters when a student may be on a co-op work term. I simulated course planning in the synthetic pathways by using a static or variable maximum number of courses chosen in each semester of the pathway. Throughout simulations, the maximum number of courses chosen in each semester was 4, 5, or 6. Static maximum simulations maintained this limit throughout the pathways while variable maximum simulations randomly chose 4, 5, or 6 each semester.

To create a sample of synthetic pathways, I simulated 3 million pathways using the static maximum and again using the variable maximum. Then, I removed simulated pathways that were invalid (more than 7 years, included part-time semesters not in the last year before graduation). In total, there were 2,128,794 valid synthetic pathways where 59% of pathways came from a static maximum number of courses in each semester and 41% came from a variable maximum. This inequality is supported by my initial testing where I only simulated pathways with varying widths as I thought it would better model how students chose courses while also including static maximum semester enrollments by chance. When only simulating variable maximums, no pathways graduated in four years and almost none graduated in four and a half years. I was able to include pathways that finished the program in four and four and a half years by simulating static maximum semester enrollments.

Of the 2,128,794 pathways, the average pathway length was 5.46 years with the most common length being 5 years. As the synthetic training data is designed to represent a real training dataset, it is important that the synthetic data is representative. Real data on pathway lengths from the ECE department were not available, however these statistics of mean and mode are plausible and examples for graduating in 4 to 7 years are represented. Details on the lengths of pathways is tabulated in Appendix D. After creating a large sample of synthetic course pathways, the next step to constructing a values-oriented dataset is to label the synthetic pathways with the findings from the value-mapping in Chapter 3 onto each pathway and develop a representation for a value-maintained approach to course selection that differentiates between the amount of maintained values present in each pathway.

4.2. Dataset Development

It was important to develop the dataset independent to the machine learning architecture as the dataset should be useable for a variety of machine learning models for the possibility to explore changes in model performance that arise from changing the machine learning architecture. To develop the values-oriented dataset, I used the results from the values survey and course topic labelling, to label the synthetic pathways. Recall the survey results for utility and cost values were quantized to numbers between 0 and 1 (for each of the 6 survey items) such that each response on the Likert-scale is separated equally. 0 represented strongly disagree and 1 represented strongly agree. I chose to use the average of all responses to represent each survey item for the constructs of utility and cost for each course. From the interest coding results, topics that occurred in fewer than 4 courses were not considered in this model as the deviations (spread) of those topics would be similar in most pathways (recall pathways are 8 to 14 semesters long).

There were 10 topics that occurred in 4 or more courses as outlined by the interest coding results in Table 15 of Chapter 3. Interest quantization for each appearing topic was chosen as $\frac{1}{n}$ where n was the number of interest topics appearing within the course (e.g. a course with 3 topics would be labelled as $\frac{1}{3}$ for each of the appearing topics). If the topic did not appear in the course, the course was labelled as 0 for that topic. This is a general assumption that is meant to identify what topics are present in a course while considering that the more topics are present the less impact each topic may have. To represent the value representation of a semester of courses, I averaged the utility and cost aspects and used maximum absolute scaling for interest to ensure each interest aspect was between 0 and 1.

With the above quantization approaches for the values of utility, cost, and interest, I labelled the synthetic pathways based on their semester-wise value representations. I then investigated the mean and standard deviation of each value representation for each pathway in the dataset of synthetic pathways using a tree data structure. By using a tree, I ensured the synthetic pathways and their sub-pathways were filtered for duplicates (as found in cases of convergent pathways, i.e. the same final semester of courses). Each node of the tree contained what courses had been taken, with the root representing no courses taken, and all pathways that follow from the previously taken courses. Figure 10 presents an example of a case where four courses need to be taken (A, B, C, and D), and only one course is taken per semester.

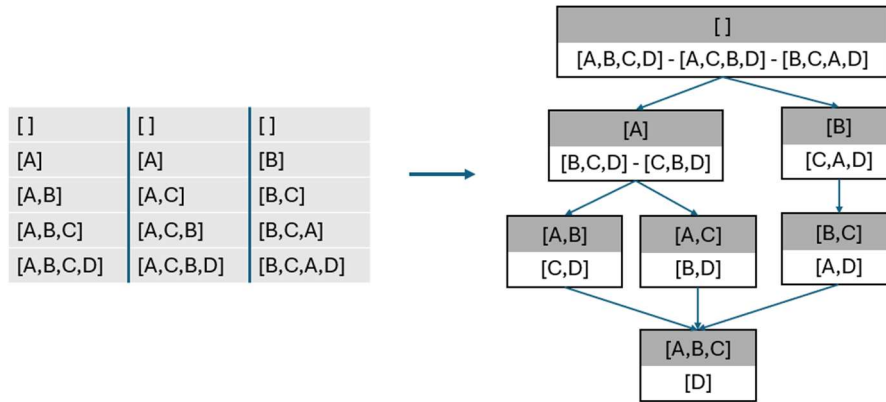


Figure 10. From pathways to a tree of unique sub-pathways

In this example, there are three distinct course pathways from zero courses to all four courses taken (commas separating each semester) on the left, and the tree of nodes on the right. In the tree, the top of the node (in grey) is a list of the courses that have been taken regardless of the order, while the bottom of the node contains all pathways to completion from the point at which the courses at the top of the node have been taken. The tree was built by taking a pathway, checking to see if it exists in the current node (beginning at the root node at the top) adding itself to the current node, then traversing to (or creating a new node if it doesn't exist) the next node based on the course taken in the first semester and repeating the process recursively (course taken in the first and second semester, etc.) until no more semesters of courses remain and you have reached the bottom of the tree. This recursive approach is similar to a sliding-window approach to slide a window of included courses taken backwards across each pathway, gradually including each semester of courses in the window. For the tree, the same approach from Figure 10 was used but instead of a single course being taken each semester, a set of courses were in each semester.

I used the sample of all synthetic pathways and sub-pathways within the tree to calculate the mean and standard deviation of each value aspect, displayed in Table 17.

Table 17. Maintained value mean and std. dev. for the synthetic course pathways tree

Value Parameter	Mean	Standard Deviation
Utility1	0.0951	0.0247
Utility2	0.0881	0.0261
Utility3	0.1120	0.0255
Cost1	0.1528	0.0333
Cost2	0.0827	0.0367
Cost3	0.0783	0.0300
Interest1 – Engineering Design Process	0.2645	0.0650
Interest2 – Mathematics	0.2162	0.0735
Interest3 – Program Design & Data Structures	0.2015	0.0568
Interest4 – Digital Systems Design	0.1423	0.0889
Interest5 – Computer Architecture	0.1551	0.0972
Interest6 – Electronics	0.1575	0.1054
Interest7 – Software Engineering	0.2283	0.0564
Interest8 – Mechatronic Design	0.1875	0.0605
Interest9 – Computer Communications	0.1990	0.0641
Interest10 – Distributed Systems	0.1535	0.0773

As the maintained value mean was calculated from the standard deviation, a lower mean represents higher maintained value. As such, Utility1, Utility2, and Cost3 on average are the most maintained values within the dataset while Interest1 has the lowest maintained value average. The standard deviation of maintained values within pathways can be used to understand the variance in maintained value representations. Interest6 has the highest standard deviation, which means that pathways are the most variability in how well maintained the value is.

I also tracked cases where an interest topic was not present in a semester as the standard deviation would be misleading if many semesters were lacking the topic (many semesters of 0 interest would appear to have high maintained value), displayed in Table 18.

Table 18. Statistics of interest topics not occurring in semesters

Interest Parameter	Zero Semester Occurrences Mean	Zero Semester Occurrences Standard Deviation
Interest1 – Engineering Design Process	1.606	1.337
Interest2 – Mathematics	1.731	1.131
Interest3 – Program Design & Data Structures	1.652	1.233
Interest4 – Digital Systems Design	3.724	1.520
Interest5 – Computer Architecture	4.090	1.740
Interest6 – Electronics	3.846	1.562
Interest7 – Software Engineering	2.011	1.928
Interest8 – Mechatronic Design	0.711	1.026
Interest9 – Computer Communications	3.340	1.969
Interest10 – Distributed Systems	3.895	2.017

It was important to consider how often topics of interest did not occur in semesters throughout pathways to properly represent maintained values. From these results, Interest5 is the topic that is most often missing from semesters in a pathway while Interest8 is often included in every semester of a pathway. I used these results to inform the level of maintained interest values in a pathway because, if the topic is missing from more than its average amount of semesters, it is not maintained throughout.

With the mean and standard deviation of each maintained value for the population of all generated synthetic course pathways, I then labelled each pathway categorically as maintaining “more”, “average”, or “less” value for each of the 16 value aspects. I computed maintaining

value as the standard deviation of the value across semesters in the course pathway, so lower standard deviation between semesters represents a higher amount of maintaining value and higher standard deviation represents a lower amount – an inverse relationship. As such, to label each pathway as maintaining “more”, “average”, or “less” value, I used 3 bins following an assumed normal distribution shown in Figure 11.

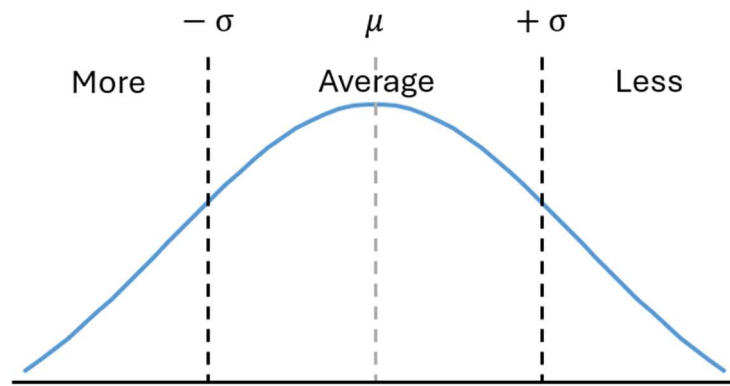


Figure 11. Maintained value binning estimated by a normal distribution

I chose these 3 bins to categorize maintained value using the mean and standard deviation of the quantified maintained values of the sample of pathways from the tree. Numerical representations of maintained values that exists within one standard deviation of the mean is considered “average”, while lower than 1 standard deviation is “more”, and higher than 1 standard deviation is “less”, as displayed in Figure 11. This approach distinguishes between the levels of maintained value throughout the dataset of synthetic pathways. The broad choice of using 3 categories aimed to distinguish between course pathways that have meaningfully different levels of maintained values. A benefit to this approach, besides its simplicity, was that it shifted the input space of the machine learning model from continuous (any fractional number from 0 to 1) to categorical (1 of

3 categories), which simplified using the model. By taking a categorical approach, users may experience higher usability by choosing between three options instead of attempting to quantify their desired values between 0 and 1.

After labelling pathways with categories of maintained values, I labelled two aspects of expectancy onto the pathways. The first aspect was the length of the pathway and the second was the maximum number of courses to appear in any given semester. I completed this process for each pathway and sub-pathway such that each training example had an expectation of time to graduation and an upper limit on the courseload.

My approach to represent maintained values within pathways with three bins and using those same bins for users to specify their maintained value preferences, aligned with the trade-offs of the balance-oriented approach to course selection (Ma et al. 2021). Furthermore, by representing expectancy as time to graduation and a courseload maximum, course pathway recommendations may support students' expectations of success. Together, by incorporating both expectancy and values into my CRT, my CRT may learn to make recommendations aligned with the course pathways planned by users through their course choice motivations.

Below is an example of the process I used to label course pathways. Table 19 displays an example of a 4-year course pathway comprised of 8 semesters that met all requirements for graduation.

Table 19. 4-year course pathway example

Semester	Course Codes					
1	ANTH 2430	ENG 1450	ENG 1460	MATH 1210	MATH 1510	Written Req.
2	COMP 1012	ENG 1430	ENG 1440	ENG 3000	MATH 1710	PHYS 1050
3	CHEM 1100	COMP 1012	ECE 2220	ECE 2262	ENG 2040	MATH 2132
4	COMP 2140	ECE 2160	ECE 3610	MATH 2130	MGMT Elec	MGMT Elec
5	CHEM 1122	COMP Elec	ECE 3740	ECE 4240	MATH 2136	SCI Elec
6	COMP Elec	COMP Elec	ECE 2400	ECE 3700	ECE 3760	ECE 3780
7	CHEM Elec	COMP Elec	ECE 3400	ECE 4600	ECE Elec	STAT 2220
8	COMP 3430	ECE 4260	ECE 4600b	ECE 4830	PHYS 2152	

By using the survey results, I calculated the average of each of the utility and cost aspects of each semester, shown in Table 20.

Table 20. Pathway example semester-based utility and cost averages

Semester	Utility1	Utility2	Utility3	Cost1	Cost2	Cost3
1	0.683	0.708	0.590	0.483	0.344	0.287
2	0.705	0.774	0.657	0.464	0.321	0.264
3	0.746	0.781	0.710	0.493	0.349	0.341
4	0.647	0.657	0.628	0.424	0.390	0.372
5	0.652	0.633	0.596	0.466	0.405	0.382
6	0.802	0.821	0.735	0.511	0.394	0.400
7	0.785	0.815	0.730	0.574	0.403	0.383
8	0.825	0.822	0.807	0.633	0.403	0.462

I then computed the standard deviation of each value aspect across the 8 semesters in the pathway. Next, I used the mean and standard deviation from Table 17 to categorize the standard deviation of the pathway values into bins of maintaining “more”, “average”, or “less” value in Table 21.

Table 21. Pathway example maintained utility and cost binning

Value Aspect	Utility1	Utility2	Utility3	Cost1	Cost2	Cost3
Standard Deviation	0.0644	0.0708	0.0716	0.0630	0.0310	0.0591
Maintained Value Bin	More	Average	More	More	More	Average

I repeated the process for the interest aspects with scaled representations in each semester in Table 22 and their maintained value binning in Table 23 and the expectancy labelling is displayed in Table 24.

Table 22. Pathway example semester interest max scaling

Semester	I1	I2	I3	I4	I5	I6	I7	I8	I9	I10
1	0.00	1.00	0.00	0.50	0.00	0.50	0.00	0.00	0.00	0.00
2	1.00	0.50	0.50	0.00	0.00	0.00	0.00	0.50	0.00	0.00
3	1.00	1.00	0.50	0.50	0.50	1.00	0.00	0.50	0.00	0.00
4	0.00	0.50	0.50	0.50	0.50	0.50	0.00	1.00	0.00	0.00
5	0.33	0.67	0.67	0.33	0.67	0.33	0.67	1.00	0.67	0.67
6	0.40	0.60	1.00	0.20	0.00	0.00	0.60	0.80	1.00	0.60
7	0.75	0.75	0.75	0.25	0.25	0.25	1.00	0.75	0.50	0.50
8	0.50	0.00	1.00	0.00	0.00	0.00	0.50	1.00	1.00	0.00

Table 23. Pathway example maintained interest binning

Value Aspect	I1	I2	I3	I4	I5	I6	I7	I8	I9	I10
Standard Deviation	0.27	0.20	0.21	0.13	0.15	0.26	0.19	0.21	0.22	0.07
Maintained Value Bin	Avg.	Avg.	Avg.	Avg.	Avg.	Avg.	Less	Avg.	Avg.	More

Table 24. Expectancy labelling for time to graduation and enrollment maximum

Time to Graduation (Semesters)	8
Maximum Enrolled Courses	6

For this example, the pathway length was 8 semesters and the maximum courses enrolled in any given semester was 6. By following this process, I was able to construct a dataset of pathways where each pathway was labelled with a bin for maintained value for each of the 16 value aspects and the 2 expectancy aspects.

Next, I created training datasets to support two formats for representing course histories. The first format was a checklist of all courses (tracking completed courses without any order) while the second format was similar to a transcript where completed courses are grouped into semesters and ordered by year. I used all of the unique synthetic course pathways through the curriculum, and their sub-pathways to create a Tensorflow (Dean et al. 2015) training dataset for each model input format to be investigated. I constructed fractional dataset sizes of 0.5%, 1%, 2%, and 5% from the full dataset to facilitate model training under hardware constraints. I constructed each partial dataset from the full dataset with the following computational conditions: using 42 as the random seed; a shuffle buffer of 200,000; and proportional sharding to achieve the desired fractional size as $\frac{100}{s}\%$ where s is the number of shards). These constants ensured that each dataset was made from the same pathways to allow for accurate comparison of training behaviour. Each dataset was separated into 80/20 training/validation splits at training time.

4.3. Resulting Datasets

In total, I simulated 2,128,794 unique synthetic pathways through the curriculum. I created many training examples from each pathway in the tree through its sub-pathways. As such, the total size of each dataset for each input format of courses (checklist or transcript) was significantly larger

(approximately 6 and 10 times larger) compared to the total number of synthetic pathways, displayed in Table 25.

Table 25. Total dataset size resulting from all synthetic pathways for each input format

Input Format	Total Dataset Size (Training Examples)
Course Checklist	13,363,413
Course Transcript	21,137,545

As the Course Checklist format simplified the representation of courses, many more convergent training examples occurred (examples where the same courses had been completed regardless of the ordering) and were not used to avoid duplicates, resulting in a much smaller dataset size compared to the Course Transcript format. Table 26 includes the sizes of the fractional datasets that were used for hyperparameter optimization, model comparison, and cross validation (discussed in the next chapter).

Table 26. Sizes of datasets used for model optimization and selection

Dataset Purpose	Input Format	% of Total	Dataset Size
Hyperparameter Optimization	Course Checklist	0.5	66,817
Hyperparameter Optimization	Course Transcript	0.5	105,687
Model Comparison	Course Checklist	1	133,634
Model Comparison	Course Transcript	1	211,375
Model Comparison	Course Transcript	2	422,750
Cross Validation	Course Transcript	5	1,056,877

With the full dataset of values-oriented pathways laid out for two types of input formats and fractional datasets created, the machine learning model was ready to be trained. In the next chapter, I present the design and implementation of my course recommendation model, and show how the values-oriented course pathways dataset was used as training data to find an optimal course recommendation model.

Chapter 5: Design and Implementation of my Course Recommendation Model

This chapter presents the third phase of this project where I designed and implemented my course recommendation model, highlighted in Figure 12.

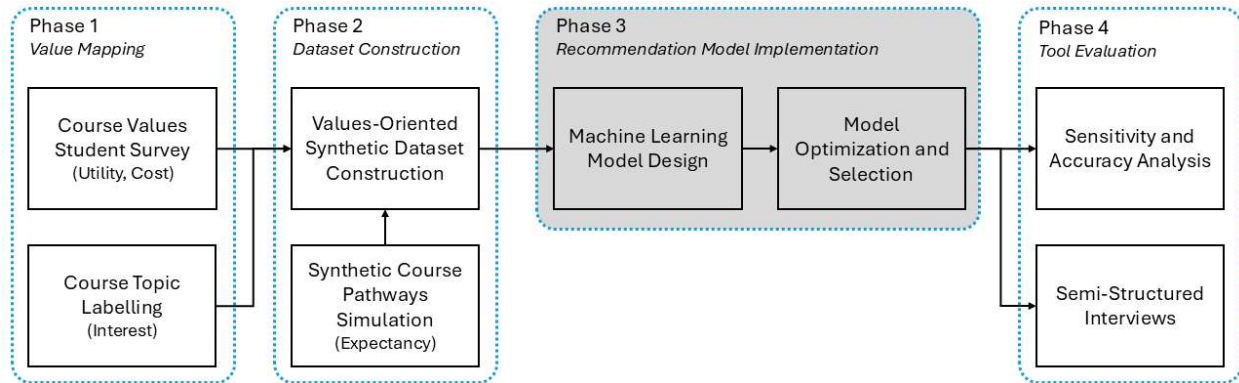


Figure 12. Phase 3: Course Recommendation Model Implementation

The design of my course recommendation model focused on the Transformer architecture (Vaswani et al. 2017). I used simple design approaches to not overcomplicate the model since my CRT was designed to be a research instrument to investigate my methodology for operationalizing social science theories of students' course choice motivations and behaviours for machine learning. As such, I did not investigate fine-grain design choices, which I left open inquiry in future research.

To find optimal architecture configurations, I used Optuna (Akiba et al. 2019) to automate the processes of searching the hyperparameter configuration space for the most accurate configurations. I investigated three input structures through automated hyperparameter optimization (HPO) and compared each model against models of the same input structure. I then

used 5-fold cross validation to select the best model of each input structure that had the highest overall performance for recommending values-oriented semesters of courses.

5.1. Tool Architecture

The architecture of my CRT consisted of an encoder-only Transformer model (BERT-based) to predict course recommendation probabilities; sampling logic to ensure only valid courses are sampled from the model's output during pathway generation; and logic to translate from user input to model input and from model output to user output.

5.1.1. Transformer Model Overview

The Transformer architecture (Vaswani et al. 2017) was created for the task of machine translation, meaning it excels at sequence prediction tasks. Thus, the Transformer lends itself nicely to course recommendation tasks and has been successfully implemented for a course recommendation tool in other research (Shao et al. 2021). The Transformer is made up of the following sequential layers: input embedding, positional embedding, attention heads (often many layers of attention heads), selecting the output, and de-embedding the output. The input embedding layer increases the dimensionality of the input data, which gives the model more room to represent underlying information. Positional embedding is then applied to represent the information to be learned from the ordering of the sequential input data. The attention heads are where the intelligence of the Transformer happens. Within the Transformer layer, context from the input data is extracted by the multi-headed attention and then information stored in the feed-forward network. After the Transformer layer(s), the final column of the output is selected

because only the next semester of courses being predicted. Then de-embedding is applied to bring the output dimensionality down to the original input dimensionality. Finally, the predicted semester of courses is output with a Softmax activation to represent a probability distribution of choosing each course.

Two options for positional encoding were discussed in the original Transformer paper, “sinusoidal” encoding and “learned” encoding (Vaswani et al. 2017). I chose “learned” positional encoding for its simplicity to implement. I also used an improvement from the initial Transformer architecture for layer normalization to increase the model’s expressivity and training convergence speed by moving the existing normalization layers from “post-norm” to “pre-norm” (Brody et al. 2023; Xiong et al. 2020; Xu et al. 2019). This design choice places the normalization from after the attention heads to before the attention heads and from after the feed forwards network to before the feed forward network. As the aim of this research project is to address the feasibility of my values-oriented approach to course recommendation, I did not want to overcomplicate the architecture by sifting through the many improvements made to the Transformer since its widespread adoption. The architecture I implemented for this project is presented as a block diagram in Figure 13 and I used the Keras (Chollet 2015) and Tensorflow (Dean et al. 2015) software libraries.

In Figure 14, the courses on the x-axis progress from preliminary-year courses on the left, followed by second-year courses, then third-year courses, fourth-year courses, and finally elective courses on the right. As there was no course history used in the example, preliminary-year courses were the priority for students to enroll in, while upper year courses that do not have prerequisites (such as ANTH 2430 and ENG 3000) and elective courses also have non-zero probabilities of being recommended. This example highlights how a preliminary-year course such as MATH 1710 should have a near-zero probability as its prerequisites are MATH 1510 and PHYS 1050 as the other courses missing prerequisites also have near-zero probability.

By training the model to predict the courses in the next semester, my CRT was able to sample from the output distribution to form a set of course recommendations for the next semester.

When successive semesters were sampled and appended to the course history, a course pathway to graduation was able to be recommended.

5.2. Designing Training Examples for the Transformer Model

Given the overall design of my CRT, the next step was to prepare my values-oriented dataset of course pathways into training examples for my Transformer model. I thought of three key aspects when designing the training examples for my Transformer model: the general characteristics of training data; representations of course histories in the input, and representations of values trade-offs and expectancy. The output representation was simpler to design than the input since the output needs to recommend the courses to take in the next semester. As such, the target output of the training example is the next semester of courses, which is already part of the dataset.

5.2.1. Characteristics of Transformer Training Data

Transformer architectures are primarily used for sequence prediction tasks, that is, tasks that involve predicting what comes next after a sequence of inputs. To train a model to predict the next output, sequential formatting must be followed when creating pairs of training inputs and target outputs as the order of the input can affect the prediction.

For example, generating text is a common use case for Transformer models. In a simple form of text generation, the start of a sentence is the input and the model predicts the next word in that sentence. A core idea of text generation using a Transformer model is that the next word generated is based on the underlying context within the given input sentence. For example, if the input is “Famous singer Michael” the output may be “Jackson” whereas if the input is “Famous basketball Michael” the output may be “Jordan”. Only one word is different between these examples, but that word provided meaningful context to change the output. Another key aspect of context for text generation is the order of words. In contrast to the above example, if the input is “Michael famous basketball” the output may be “player” instead of “Jordan”. This prediction may be more grammatically correct (despite not being full sentences). The difference between these two examples is whether or not positional encoding was used to train the model to learn the ordering of words in the input, rather than only making predictions based on what words are in the input.

For my course recommendation model, the context provided in the input is the maintained value trade-offs and a history of completed courses completed. Values, in the context of EVT, do not inherently have an ordering whereas, in the context of course pathways, courses completed do have an ordering in the form of prerequisites. As such, it was important for training examples to

properly represent courses taken in an orderly manner through the use of positional encoding in the Transformer architecture.

5.2.2. Representation of Courses Completed

To represent the courses a student user has completed, I represented each course at the input level as either being either 1: taken, or 0: not taken. Given that I modelled UM's computer engineering curriculum as 60 different courses (including core courses, and variations of what electives are taken to satisfy program requirements), I used a 1×60 multi-hot vector, where each element is either a 0 (course not in the semester) or a 1 (course is in the semester), to represent courses completed. This vector on its own could track the courses a user has completed like a checklist where initially all elements are 0 and become 1 when completed. Alternatively, a matrix of $M \times 60$ multi-hot vectors, where M is the maximum number of semesters a student can enroll, tracks what courses have been completed and when courses were completed (like an academic transcript). As these two approaches (checklist and transcript) may have different benefits or drawbacks to modelling, and are not mutually exclusive, I also investigated a third approach of combining both the single vector check list and the matrix of semesters. Given sufficient model size and training regimen, the input structure with the most meaningful information to learn from should result in the highest modelling performance. Figure 15 displays simplified examples of the three representations for courses taken investigated in this research. Throughout this thesis, the models resulting from these formats are referred to as Courses Taken Vector (CTV), Semester Sequence, and Hybrid respectively.

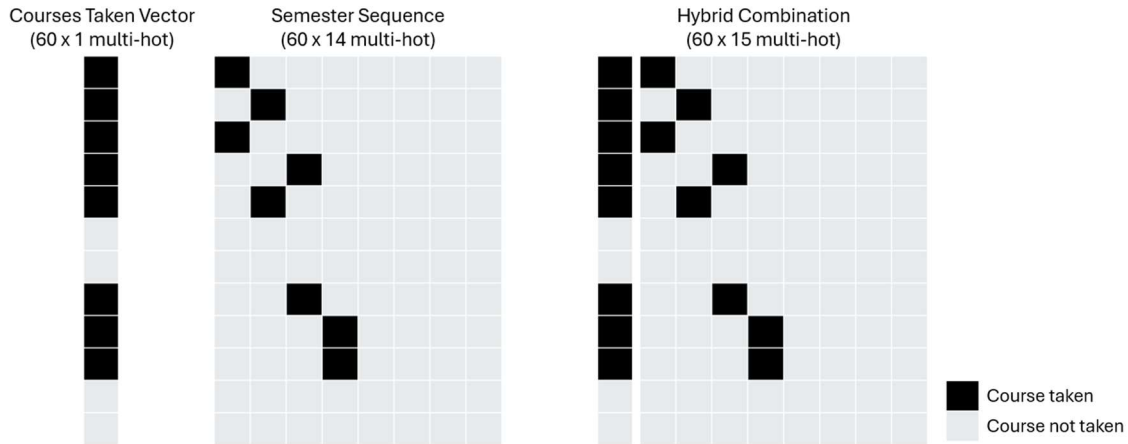


Figure 15. Input representations of courses taken

5.2.3. Representation of Student Value Trade-Offs

Being able to represent computer engineering students' value trade-offs is a core decision in the design of my course recommendation model since the aim of this tool, and my overarching methodology, was to recommend courses based on the value trade-offs of computer engineering students. Expectancy was represented by factors defining how many semesters remain until graduation, and the maximum number of courses taken in each of those semesters. As such, I represented the length of pathways and sub-pathways categorically from 1 to 14 remaining semesters, given a 7-year maximum graduation time, as a 1 x 14 vector. Then, I represented the maximum courses enrolled in any given semester as either 4, 5, or 6, courses, as enrolling in more than 6 courses required special permission within the program, as a 1 x 3 vector.

I represented interest, utility, and cost similarly to each other for each of their underlying aspects (10 interest topics, 3 utility items, 3 cost items). Since I designed the values-oriented dataset around maintained value trade-offs, the mean and standard deviation of each aspect was used to categorize how maintained each value aspect was into 3 bins, as described in Chapter 4.2. As

such, the smallest building block for the utility, cost, and interest representations was a 1 x 3 vector to categorically represent “more”, “average”, or “less” maintained value. Figure 16 shows how I used this 1 x 3 vector by filling the vector with 0’s and using a 1 to indicate a category of maintained value.

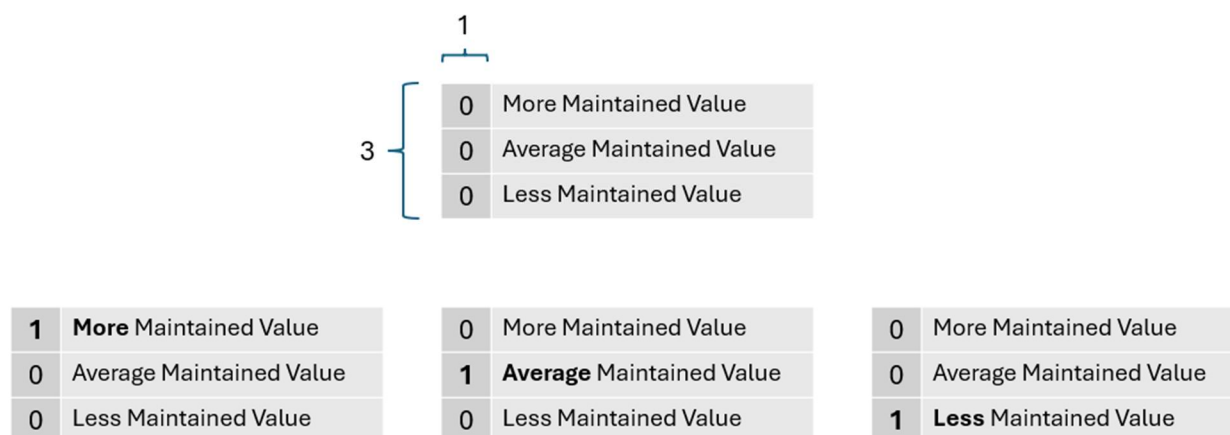


Figure 16. Categorical maintained value vectors

This approach allows for the first index to represent “More Maintained Value”, the second index to represent “Average Maintained Value”, and the third index to represent “Less Maintained Value”. A single vector filled with 0’s and a single 1 it is commonly referred to as a “one-hot” vector whereas a vector filled with 0’s and many 1’s is referred to as a “multi-hot” vector. After I applied the 1 x 3 one-hot vector to each of the three utility aspects, I concatenated the 3 utility vectors into a 1x9 vector as shown in Figure 17.

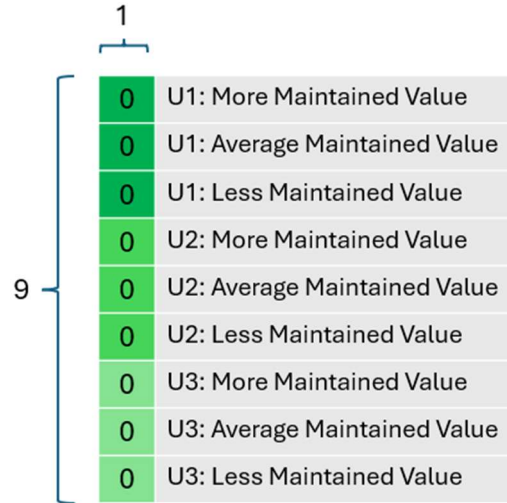


Figure 17. Concatenated 1 x 9 utility input vector

In this 1 x 9 vector, three 1's are used to indicate the category of maintained value for each of the three aspects of utility, resulting in a multi-hot vector. With this approach, I represented each aspect of utility, cost, and interest (3 utility, 3 cost, 10 interest) categorically as either “more”, “average”, or “less” maintained value through a 1 x 3 vector. I then concatenated all the maintained value vectors together in a single multi-hot vector of size 1 x 48. Figure 18 presents a simplified diagram of how the EVT values are represented in the input space for the model.

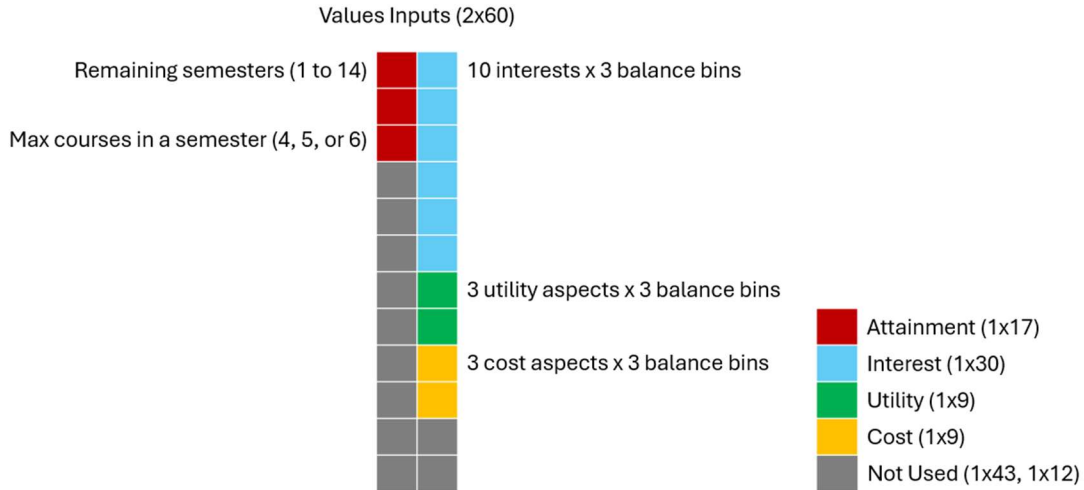


Figure 18. Input vectors space for motivational value aspects

Due to the constraint of sequence width based on the number of courses modelled in the program (60), and for simplicity, the values were in a separate column vector from expectancy. One vector contained the representations of the two aspects of expectancy (1 x 17) while the other vector contained interest, utility, and cost (1 x 48). Splitting the EVT values and expectancies into two separate columns in the input matrix meant that position was given meaning (due to positional encoding), however, by placing these two vectors at the same positions in every training example, the ordering will have the same impact on every training example. Additionally, as my course recommendation model is BERT-based, both columns will attend to each other, rather than a decoder-based model that would prevent this through causal masking (only attending backward). As such, the ordering of expectancy and values within the input does not negatively impact training the model.

5.2.4. Model Input Formats

I investigated the 3 input formats for courses taken, each concatenated with 1 input format for EVT preferences, for which format of courses taken had the highest performance. These input formats are visually simplified in Figure 19.

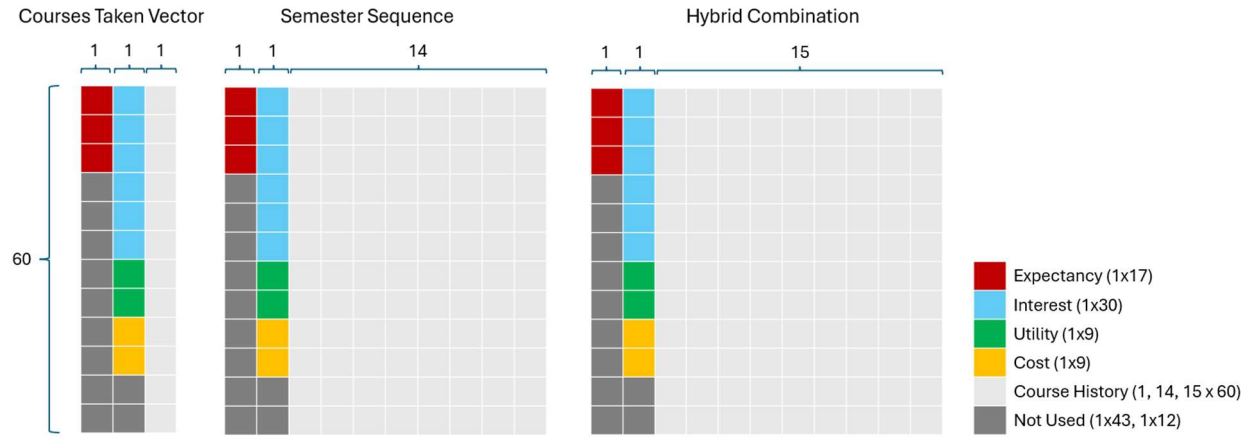


Figure 19. Model input structures including EVT preferences and completed courses

Each of the 3 input formats were sequential in the shape of 3 x 60, 16 x 60, and 17 x 60 for Courses Taken Vector, Semester Sequence, and the Hybrid Combination respectively. Column 1 contains the representation for expectancy and column 2 contains the representations of maintained value trade-offs for interest, utility, and cost. The remaining columns contain the representation of completed courses. After defining the model inputs, the next section presents the training loss function.

5.3. Loss Function

The learning part of machine learning consists of making changes to the internal weights of the model to reduce the loss of information. Various loss functions can be used to calculate the loss of information during training. Ciampiconi (2024) categorized common loss functions based on the type of model and application. I used binary cross-entropy alongside a weighted Kullback-Leibler Divergence (KLD) for the training loss function that are at the intersection of loss functions used for classification and generative models with a probabilistic output (Ciampiconi et al. 2024). For the task of course recommendation, classification occurs when the model predicting the courses in the next semester given the training data however, the CRT is also designed to be generative to recommend course pathways that are not present in the training data. For the CRT to generate a variety of pathways given the same input it must be probabilistic, instead of deterministic, such that there is a probability that courses will appear in the next semester. Courses that are not the target recommendation may still be chosen but with a much lower probability.

The loss function I used follows the following mathematical description. With C possible courses (in my CRT $C = 60$), a semester of courses representing the ground truth (from the training data) is

$$\mathbf{y} = [y_1, y_2, \dots, y_C] \quad (1)$$

where

$$y_i = \begin{cases} \mathbf{1} & \text{if course } i \text{ is in the target semester} \\ \mathbf{0} & \text{otherwise.} \end{cases} \quad (2)$$

The model predicts

$$\hat{\mathbf{y}} = [\hat{y}_1, \hat{y}_2, \dots, \hat{y}_C], \quad 0 < \hat{y}_i < 1, \quad (3)$$

where $\hat{y}_i$ in Equation 3 represents the predicted probability that course i occurs in the next semester. It follows from Equation 1 and Equation 3 that the loss associated with the binary cross-entropy (Ciampiconi et al. 2024) between the ground truth and the prediction is

$$\mathcal{L}_{BCE} = -\frac{1}{C} \sum_{i=1}^C [y_i \log(\hat{y}_i) + (1 - y_i) \log(1 - \hat{y}_i)]. \quad (4)$$

The loss function for binary cross-entropy (Equation 4) penalizes the model for predicting low probabilities for target included courses and for predicting high probabilities for target unincluded courses. These penalties were averaged across all C courses.

To compare the ground truth and prediction as probability distributions I normalized $\mathbf{y}$ from Equation 1 and $\hat{\mathbf{y}}$ from Equation 3 as

$$p_i = \frac{y_i}{\epsilon + \sum_{j=1}^C y_j} \quad q_i = \frac{\hat{y}_i}{\epsilon + \sum_{j=1}^C \hat{y}_j}, \quad (5, 6)$$

where ϵ is a small positive constant to avoid divisions by 0. The sum of both p_i and q_i are 1 as they are both probability density functions. It follows from Equation 5 and Equation 6 that the loss function for the Kullback-Leibler Divergence between the two (Ciampiconi et al. 2024) is defined as

$$\mathcal{L}_{KL} = D_{KL}(p \parallel q) = \sum_{i=1}^C p_i \log\left(\frac{p_i}{q_i}\right), \quad (7)$$

where the loss from the Kullback-Leibler Divergence (Equation 7) penalizes predicted outputs where correct courses have low probability, incorrect courses have high probability, and correct courses do not have equally high probability. Then, I defined a hyperparameter λ to control the contribution of $\mathcal{L}_{KL}$ to the total loss function. The total loss function is

$$\mathcal{L}_{total} = \mathcal{L}_{BCE} + \lambda \mathcal{L}_{KL}$$

$$\mathcal{L}_{total} = -\frac{1}{C} \sum_{i=1}^C [y_i \log(\hat{y}_i) + (1 - y_i) \log(1 - \hat{y}_i)] + \lambda \sum_{i=1}^C p_i \log\left(\frac{p_i}{q_i}\right). \quad (8)$$

The loss function for my ML model (Equation 8) was the sum of the binary cross entropy loss from Equation 4 and a weight (λ) applied to the Kullback-Leibler Divergence from Equation 7. Next, I present the training metrics I used to evaluate the models trained with this loss function.

5.4. Training Metrics

To evaluate training performance, I viewed three metrics of recall@6, Kullback-Leibler Divergence (KLD), and area under the receiver operating characteristic curve (AUC). Recall@6 was the primary metric used for training validation, while KLD and AUC were tracked and inspected to assist in the model selection process. Recall@K (also known as top-K recall) and AUC are popular performance metrics for recommender systems (Bauer et al. 2024; Silveira et al. 2019). KLD is not as often used in recommendation systems but can provide additional insights on the learning of the output probability distribution (Ricci et al. 2011).

Recall@K is a classification metric that represents the proportion of positive outputs within the K highest probability outputs (Ricci et al. 2011; Silveira et al. 2019). I chose $K = 6$ over $K =$

5 and $K = 4$ because the maximum courses a student takes in a semester in the dataset is six courses. Since not all semesters contain six courses in the dataset $K = 4$ will result in an upward bias in accuracy evaluation for example semesters with only 4 or 5 courses because it checks the 6 highest probabilities together. Additionally, $\text{recall}@4$ would result in a downward bias from examples with 5 or 6 courses. In the case of evaluating based on the 4 highest probabilities, there is no way to distinguish where the 1 or 2 courses, in the cases of a semester with 5 or 6 courses, outside of the 4 highest are in the probability distribution. This means that it could be the 5th highest or the 60th highest and $\text{recall}@4$ would be the same. As such, by looking at the 6 highest probabilities for accuracy, all outputs are accounted for in all cases of 4, 5 or, 6 courses in a semester.

The area under the receiver operating characteristic curve (AUC) is a metric used to evaluate the ranking ability of the model (Fawcett 2006; Ricci et al. 2011). AUC is a helpful ranking metric that looks at the probability of correct outputs being ranked above incorrect outputs. As such, AUC approaches ranking differently than $\text{recall}@K$ but both are considered complimentary to each other. AUC lends itself nicely to assess a model trained on data with imbalanced output data (Richardson et al. 2024). Imbalanced output data simply means that the occurrences of the output classes (or in my case output courses) are not uniformly distributed, which is the type of dataset I used during training. This imbalance is due to 4 to 6 courses being the targets in the next semester and since there are 60 options to recommend from, the occurrences of the same 4 to 6 courses is also imbalanced. Thus, AUC is a helpful metric to view to gain insights on the ranking ability of my Transformer model.

Kullback-Leibler Divergence (KLD) is a metric used for comparative evaluations of probability distribution functions (Ricci et al. 2011). KLD is not a true distance metric but is useful for

comparison. KLD has also been used to train variational autoencoders (another machine learning architecture) for the task of sequence prediction (Prokhorov et al. 2019). KLD is often used in machine learning when training a probabilistic model as it tasks the model to learn the output distribution instead of only learning whether the output is correct or incorrect (Ricci et al. 2011). As such, it is a supportive metric to aid in calculating loss and to gain insight on the modelling of the output probability distribution.

5.5. Initial Training Pseudo Validation on Small Datasets

Before beginning model optimization, I first performed a pseudo validation of each of the input structures for the machine learning model. This initial pseudo validation is effectively a check to see if the training dataset contains learnable information under the assumption that EVT modelled course choice motivations contain enough learnable information to recommend values-oriented courses. To initially validate the ability of the models to do any learning from the training data, small datasets (generated from 10,000 synthetic pathways) were created for each input type and used with generic hyperparameters for each model. This approach enabled a rough approximation of whether the training data contained learnable information for modelling. If results were only marginally better than chance, this approach would reveal that change, and potential redesign, would be required for the input format or the underlying values-oriented information in the dataset.

5.6. Model Optimization and Selection

After successful pseudo validation of the model learning to recommend using the values-oriented dataset, I began the model optimization and selection process. Model optimization and selection followed a sequential process, shown in Figure 20, with the goal of identifying the best model for recommending values-oriented semesters of courses that will lead to generating values-oriented course pathways.

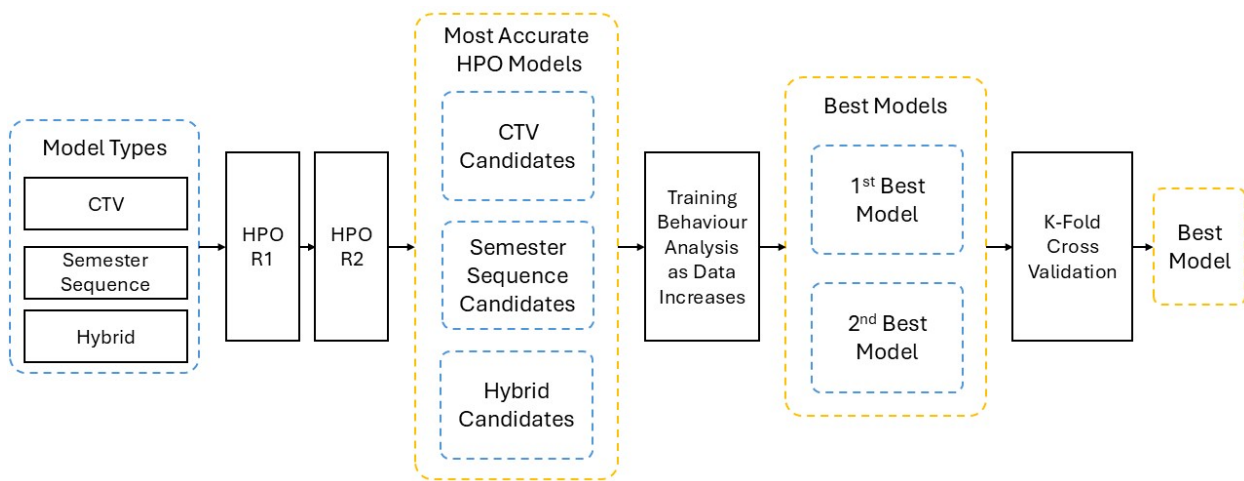


Figure 20. Model optimization and selection process

First, automated hyperparameter optimization (HPO) was used to find the best hyperparameter configurations for each of the 3 model types (Transformer models with different input formats). Given the constraints of the training hardware, HPO was divided into two parts. First a broad search space for the numbers each hyperparameter can take on (HPO round 1 / R1). Second, based on the results of the broad search, a refined search space was investigated (HPO round 2 /

R2). The most accurate models of each model type of each of the two HPO searches were then compared to find the best model of each type and then finally the best model overall. This selection process can be thought of as similar to a playoff bracket. For dataset sizes used throughout this process, please see the list of resulting datasets in Table 26 in the Chapter 4. All datasets were split into 80% training and 20% validation.

5.6.1. Hyperparameter Optimization

There are many approaches to hyperparameter optimization (HPO) (Yang and Shami 2020). Some simple methods, as described by Yang and Shami (2020), are model-free algorithms consisting of grid search, and random search. Grid search consists of defining discrete ranges of hyperparameters and testing each possible configuration. However it has been shown that random search can be just as effective without being as time consuming as grid search (Bergstra and Bengio 2012). Random search involves similarly setting up a search space or search grid and randomly sampling hyperparameter configurations. Although random search can be better than grid search, automated approaches were developed to better optimize hyperparameters and be effective on continuous hyperparameters (Yang and Shami 2020). Automated HPO methods involve using a separate model to find the areas within the search space with the highest likelihood of producing a more accurate result. Many software libraries exist for automated HPO (Shekhar et al. 2022; Yang and Shami 2020). In my search for a suitable HPO library, I came across Optuna (Akiba et al. 2019), which has been used as a simple HPO framework in peer

reviewed studies involving applied machine learning (Lai et al. 2024; Nam and Kim 2025).

Comparative research of HPO frameworks (Shekhar et al. 2022) and HPO algorithms (Sipper 2022) have demonstrated that Optuna is an effective HPO framework for research applications.

5.6.2. Optuna

Optuna (Akiba et al. 2019) is a HPO framework that allows for efficient searching of hyperparameter configurations within a defined search space on any type of machine learning model. Optuna searches the range of hyperparameter choices by training models, storing their performance, and selecting future configurations based on the performance of other trials. This effectively builds a performance function to iteratively estimate which configurations are likely to result in higher performance. With sufficient trials, Optuna is able to broadly probe the search space to ensure coverage and then converge using the configurations with the highest performance. In applied research studies for disease prediction (Lai et al. 2024) and landslide susceptibility (Nam and Kim 2025) Optuna was used for HPO with 50 trials. Another study investigated hyperparameters on a variety of machine learning algorithms using Optuna also with 50 trials (Sipper 2022). The number of trials is important since Optuna fits a performance model to the search space to identify promising search areas to focus on in future trials. If too few trials are completed, the search space might not be properly investigated, while too many trials may have diminishing returns. This understanding of Optuna guided how I used it for my application with the Transformer model.

The Transformer architecture hyperparameters I investigated were:

1. the size embedding dimension,
2. the number of sequential Transformer encoder layers, and
3. the number of attention heads within each Transformer encoder layer.

I also investigated hyperparameters independent of the Transformer architecture for:

4. the rate of machine learning,
5. a weighted portion of the loss function, and
6. the batch size of training data.

I used the Optuna library (Akiba et al. 2019) to perform automated hyperparameter optimization (HPO) to search for hyperparameter configurations with the highest validation-set top-6 accuracy (recall@6). Automated HPO is the industry standard with the highest performing optimization tools using machine learning techniques to guide HPO (Yang and Shami 2020). As this is a pilot study and the machine learning model was a research instrument, the highest performing methods were not required to show methodological feasibility. With this in mind, simple approaches such as Grid Search or Random Search (Bergstra and Bengio 2012; Yang and Shami 2020) are easy to implement but can take a large amount of time, much of which may be wasted searching unpromising configurations. Optuna allows for simpler and more efficient HPO than Grid or Random Search. To perform HPO, first, I defined a broad search space that Optuna used to search for high performing configurations of all three models on a small dataset. Then, using the results from the initial search, another search space was defined to be more refined by constricting the search space parameters for each of the three model types.

5.6.3. Broad Initial Hyperparameter Search Space

I defined a broad initial search space using ranges for Transformer architecture hyperparameters based on Transformer HPO research (Xu et al. 2025a). However, the maximum bounds were limited by the hardware I was using, locally training the models on a Nvidia RTX 3060Ti GPU. Optuna searched the defined space and trained each model using 0.5% of the full dataset for 25 trials, using an 80-20 train-validation split. Although the literature uses 50 trials (Nam and Kim 2025; Sipper 2022), hardware constraints prevented this. As the aim of this research is to evaluate the methodological feasibility of using EVT-modelled motivation for course recommendation, the 50 trials used for fine-tuned applications were not necessary to explore the performance of my approach with an understanding that there will be room for improvement.

A small fraction of the total dataset was used for HPO. A small dataset can be used to for exploratory findings of hyperparameters in less time than a total dataset since the aim of HPO is to discover what configuration best learns to generalize the dataset. Each of the 3 model types were investigated with the same training data, search-space, and number of trials. Table 27 presents the parameters being optimized, each parameter's minimum and maximum, and its respective step size for discrete hyperparameters.

Table 27. Initial hyperparameter search space

Parameter	Min	Max	Step Type (step size)
Learning rate	10^{-6}	10^{-3}	Continuous
KLD weight	10^{-3}	10^0	Continuous
Embedding dimension	128	512	Discrete (2^n)
Transformer layers	1	8	Discrete (1)
Transformer heads per layer	2	16	Discrete (2^n)
Batch size	16	128	Discrete (2^n)

5.6.4. Refined Hyperparameter Search Spaces

Based on the results from the initial search, it was apparent that convergence occurred for some hyperparameters, but this convergence varied by model type. Optuna was used with the refined search-spaces (HPO Round 2) for another 25 trials for each model type. This allows for a more focused search of hyperparameter configurations for each model type, displayed in Appendix E. When a hyperparameter converged to either the upper or lower bound of the search space, the search space was expanded so long as it still fit on the training hardware. It is important to note that due to the performance of the CTV model after the initial HPO, a refined search space was not created.

5.6.5. Comparison of Optimized Models

From the results of the broad search hyperparameter optimization (HPO Round 1), it was clear that the model using a single vector for course representation (CTV model) was significantly less accurate than the other two models, so I did not investigate it further. For the Semester Sequence

and Hybrid models the highest validation accuracy from the broad search and the informed search were then retrained using larger datasets to see which of the two models trained better. These models were trained on 1% and then on 2% of the total dataset (twice and four-times the size of the HPO dataset) to investigate how training behaviour could change as the size of the dataset increases. Comparisons were made between models of the same input type to determine the best model, considering accuracy and ranking ability, for each model type.

5.6.6. Model Selection Using K-Fold Cross Validation

K-fold cross validation is commonly used to choose between models according to a survey of cross validation procedures for model selection (Arlot and Celisse 2010). I performed K-fold cross validation by splitting the training data into K approximately equal parts, using $K - 1$ parts for training and using the remaining 1 part to validate the training. This was performed $K = 5$ times and training results were calculated for the average and standard deviation of performance from each fold acting as the validation data while the rest was used for training. K-fold cross validation aims to limit the effects of having a lucky set of validation data that skews the training performance results. In this research, I used K-fold cross validation to inform my choice between the highest performing Semester Sequence and Hybrid models. Using 5 folds is considered the standard as significant improvements are seen up until $K = 5$ with diminishing returns thereafter (Arlot and Celisse 2010; Fushiki 2011).

5.7. Machine Learning Model Results

The aim of the machine learning process was to determine the best performing ML model for recommending values-oriented course pathways. This meant that of the three different formats of structuring the input data (CTV, Semester Sequence, and Hybrid), one needed to be identified as better than the other two. Hyperparameter optimization (HPO) was divided into two parts, the initial broad search and the refined search, leading to the results of each model type also being divided in the same manner. To compare the performance of the best models of each type, dataset sizes were increased to twice and four-times of that what was used during HPO and the training behaviour provided insights on which model was more capable. In this section I discuss the K-fold cross validation results that highlight the singular best model for integration into the course recommendation tool. The results from the HPO and selection process are highlighted in Figure 21.

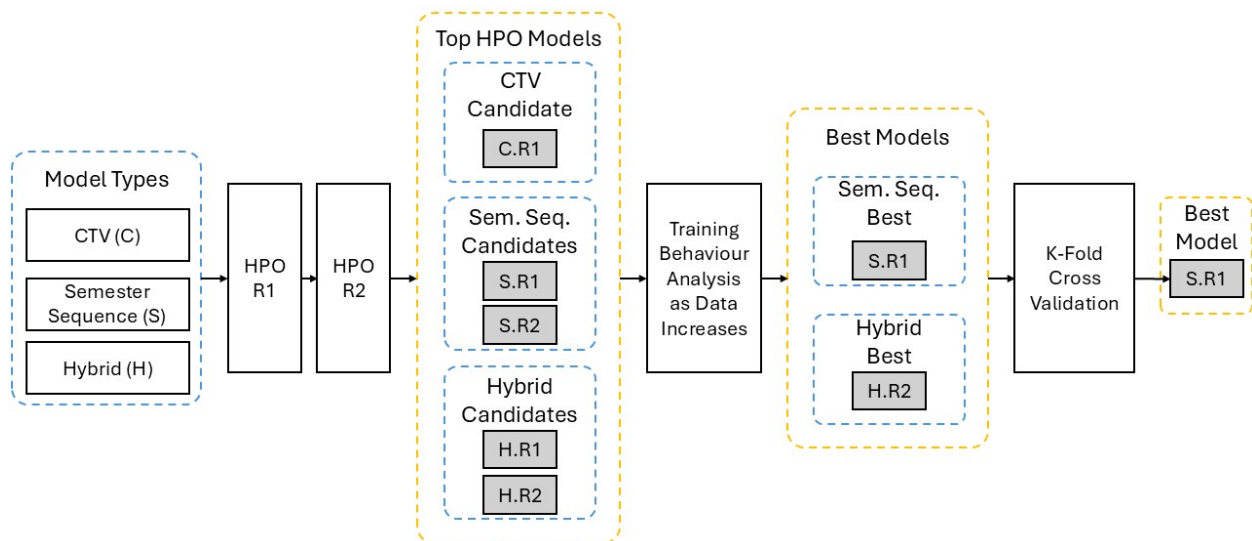


Figure 21. Machine learning model selection process and results

For brevity, Appendix E contains a detailed process of my HPO approach, which resulted in two high-performing models.

5.8. Cross Validation Results

The final part of selecting the best model was performing K-fold cross validation to compare the training behaviour of the best Semester Sequence and Hybrid models when different parts of the training dataset are used for training and validating performance, highlighted in Figure 22.

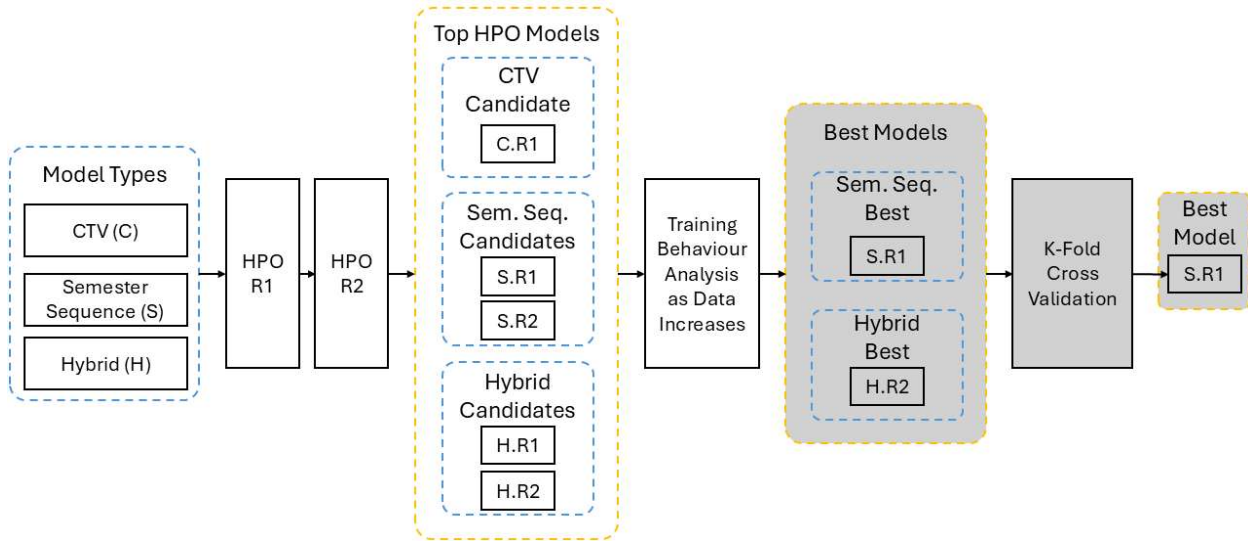


Figure 22. Cross validation portion of the model selection process

configurations of the best Semester Sequence and Hybrid models to compare and present my interpretation of how the differences affect each model. A summary of the configurations of the best Semester Sequence and Hybrid models that were compared through cross validation are displayed in Table 28.

Table 28. Summary of best models

Parameter	Best Semester Sequence	Best Hybrid
Learning rate	3.147e-4	2.167e-4
KLD weight	0.7900	2.682e-3
Embedding size	512	512
Number of Layers	6	5
Heads per layer	4	8
Batch size	64	16
Total trainable parameters	37,879,868	52,586,044

Both models, have a similar learning rate, use the same embedding size and have a similar number of Transformer layers. The first major different is the magnitude of the KLD weight used in the loss function of each model. The Semester Sequence model has a much higher KLD weight compared to the Hybrid model, which means that the Semester Sequence model likely has better modelling of the output probability distribution due to KLD's larger contribution during training. The second difference is that the Hybrid model has twice the number of attention heads per layer compared to the Semester Sequence model. By having more attention heads, the Hybrid model had more capacity for learning to extract context from the input. In addition, the increased number of attention heads in the Hybrid model is the major contributor for the difference in the number of total trainable parameters. The difference of 15 million parameters is a 38.8% increase in model size from the smaller Semester Sequence model to the larger Hybrid model. More parameters means, in general, more capacity for learning. However, the larger the model, the more time it takes to train and it uses more energy to make predictions. The difference in training batch size may suggest that the Semester Sequence model has more room for generalization by training with more examples before updating its internal weights. The plotted results from the 5-fold cross validation of the Semester Sequence and Hybrid models using 5% of the dataset is shown in Figure 23.

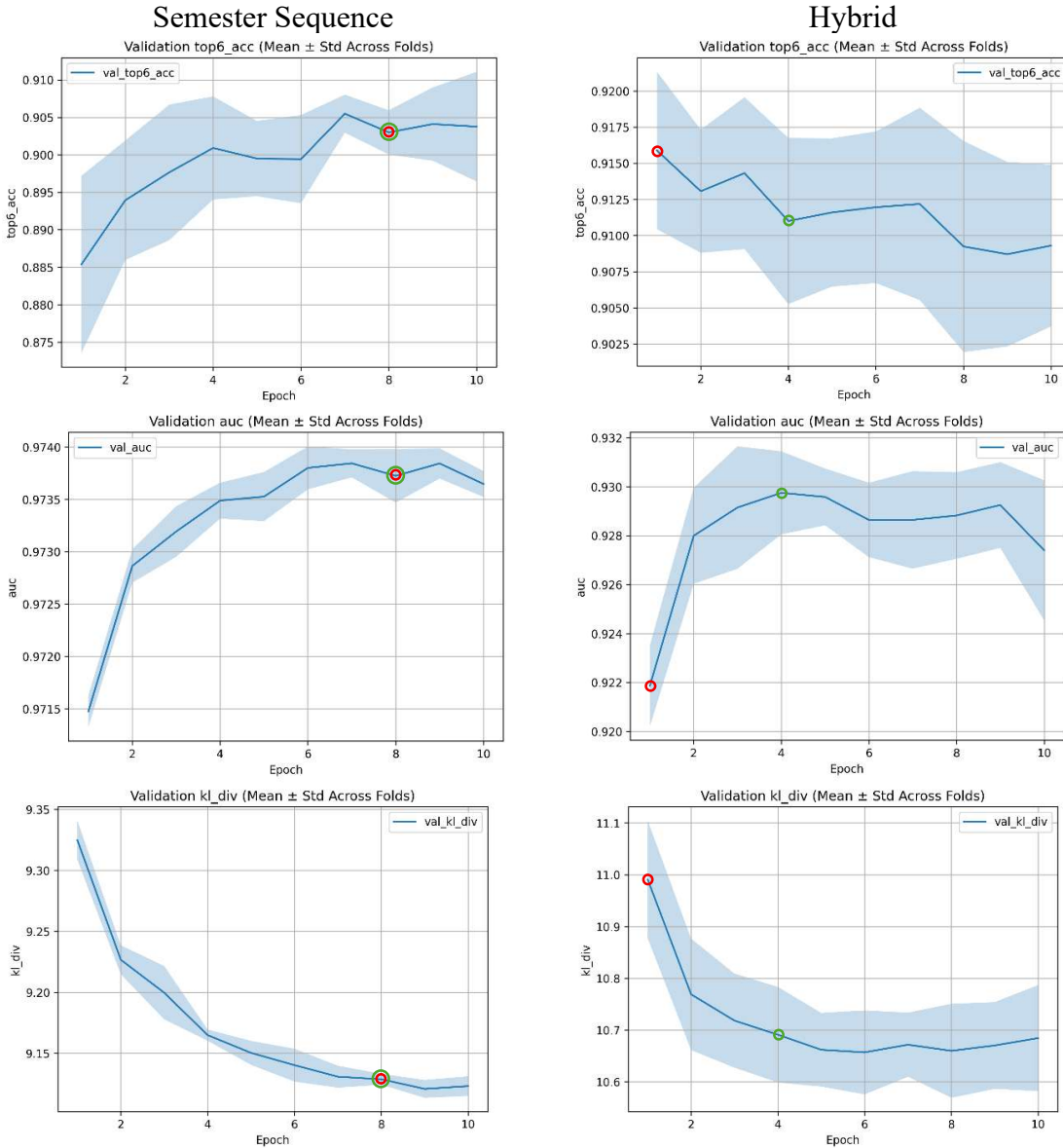


Figure 23. 5-fold cross validation mean (line) and 1 standard deviation (area) for Semester Sequence (left) and Hybrid (right) models

Both models had similar maximum values of average recall@6 (0.906 and 0.916), the Semester Sequence model had a much higher maximum average AUC (0.973 compared to 0.929) and a lower minimum average KLD (9.12 compared to 10.67) and). These results indicate that with similar validation accuracy, the Semester Sequence model represented the output probability

distribution better than the Hybrid model and the Semester Sequence model ranked positive outputs better than the Hybrid model. Although these average minimums and maximums were just presented independently for each metric, it is important to make the decision of which model is best based on the results of the best epoch of training. In Figure 23, the red circle highlights the metrics of the epoch with the highest recall@6 while the green circle highlights the metrics with the highest AUC. In the case of the Semester Sequence model, both of these cases take place on epoch 7 with recall@6 of 0.906, AUC of 0.973, and KLD of 9.13. For the Hybrid model, the red circle (epoch 1) highlights a recall@6 of 0.916, AUC of 0.922, and KLD of 11.0 and the green circle (epoch 4) highlights a recall@6 of 0.915, AUC of 0.930, and KLD of 10.7. Of these two epochs, the Hybrid model generally has better performance when choosing based on the highest AUC as there is only a small decrease in recall@6 while both AUC and KLD improve. When comparing between the best Semester Sequence metrics and the Hybrid green circle metrics, the Semester Sequence model has a slightly worse recall@6 while also having a better AUC and KLD. As such, I chose the Semester Sequence model as the best model to be integrated into my CRT since it has similar ability to be accurate compared to the Hybrid model while being better able to rank positive outputs and model the output probability distribution.

5.9. Best ML Model Output Examples

In this section I compare examples of the output distribution of the best model with different inputs value input and different course histories. Figure 24 displays the 20 courses with the highest probability when utility is “more” maintained and all other values are “average” maintained compared to when cost is “more” maintained and all other values are “average”

maintained. In both cases, no course history was used, which meant that the recommendation is for the first semester.

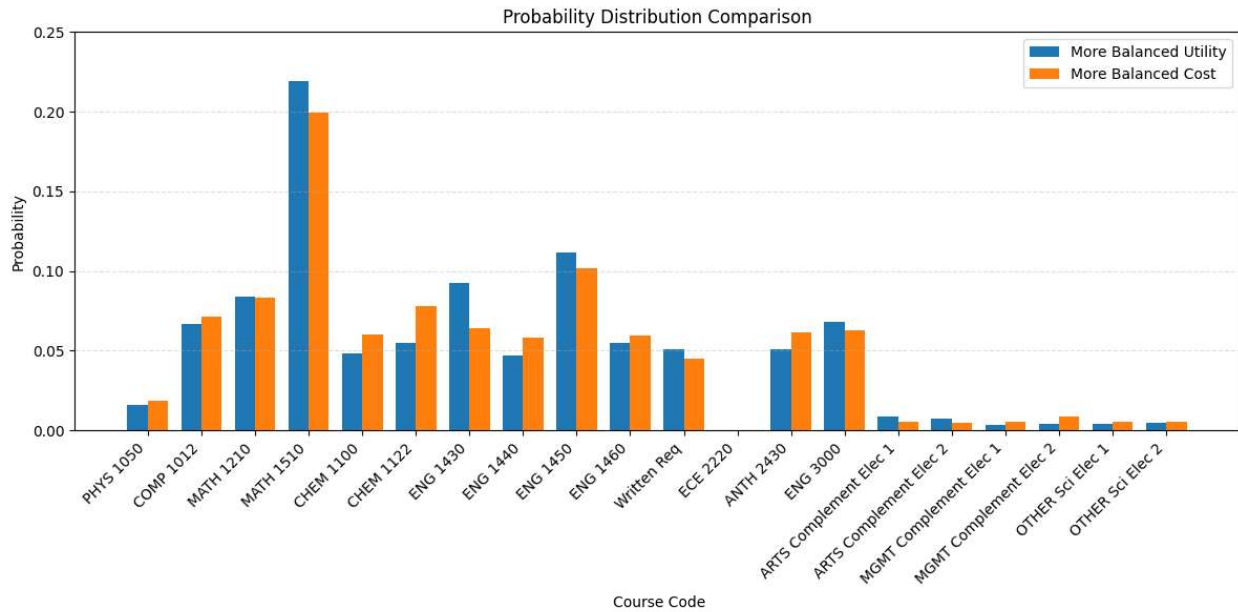


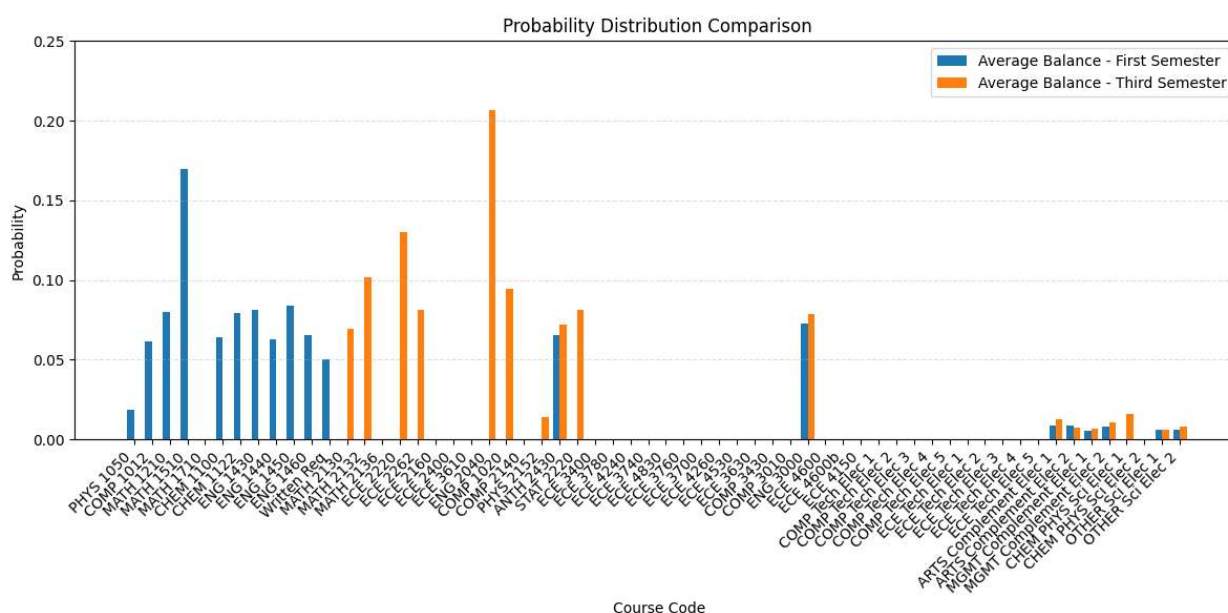
Figure 24. Utility and cost trade-off prediction comparison for semester 1

In both cases, preliminary year courses are among the highest probabilities. Additionally, there are small changes in the probabilities based on whether utility or cost is “more” maintained. For example, when utility is “more” maintained, MATH 1510 has a higher probability than when cost is “more” maintained. Another example that highlights the opposite is CHEM 1122, which has a higher probability when cost is “more” maintained compared to utility. The comparison between the maintaining utility or cost demonstrated the model was able to learn to adjust the output distribution based on maintained value trade-offs throughout course pathways.

In either case of “more” maintained utility or cost, MATH 1510 has the highest probability with a large margin of 0.1 from the second highest probable course of ENG 1450. This aligns with my

discussion of preliminary-year course planning in Chapter 3 as MATH 1510 must be taken in the first semester if a student aims to complete all preliminary-year courses in their first year.

Another key aspect of the model was how the probability distribution changes as courses are completed. Figure 25 displays the output distribution for the first and third semesters (beginning of preliminary and second years) and Figure 26 for the fifth and seventh semesters (beginning of third and fourth years).



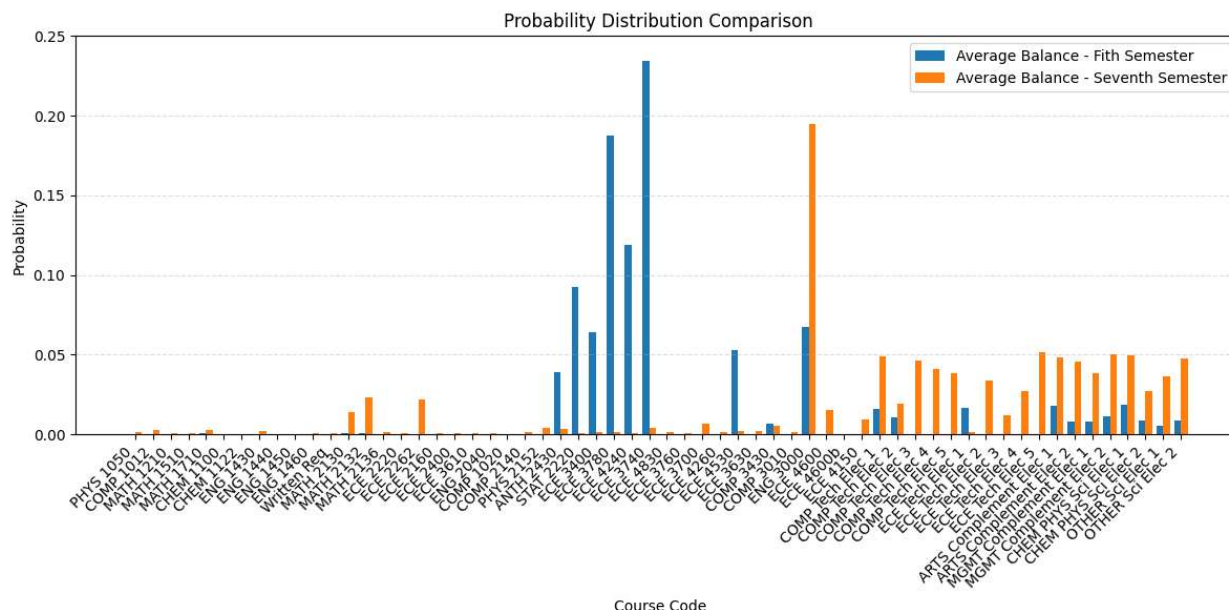


Figure 26. Output probability distribution comparison between fifth and seventh semester

In these four examples, the course history matches the recommended course pathway of the course flowchart in Appendix A. There is a clear progression of the input distribution from recommending preliminary-year courses to second-year courses in Figure 25, and from third-year courses to fourth-year courses in Figure 26. These results demonstrate that prerequisite relationships were learned by the model to adjust course recommendations.

5.10. CRT Integration

To integrate the machine learning model into my CRT, I implemented sampling logic to avoid invalid choices during course recommendations, and an interface to receive human-readable inputs and display course pathway recommendations. Since course selection has many restrictions based on course offerings and program structure, I added restrictive sampling logic. The sampling restrictions improved the sampling efficiency of by preventing invalid course

choices (recommending a course the user cannot enroll in) that would have otherwise resulted in needing to resample, which slows recommendation generation. I implemented this logic in the form of rules for recommending courses such as satisfying course prerequisites, allowing corequisites, checking the terms each course is offered in, and avoiding courses that had already been completed. As the output of the model is probabilistic, a course that should not be taken may have a very low chance of being sampled, even without placing any restrictions on the sampling. However, it is more efficient to remove invalid recommendations through these rules to avoid unnecessary resampling.

I designed the interface to be able to specify a course history, which values should be “more”, “average”, or “less” maintained throughout the recommended course pathway, expectations of the maximum number of courses in each semester, and the expectation of the minimum number of semesters to complete the course pathway. Figure 27 displays an example of using the interface format.

```

previous_courses_taken = [
    ["MATH 1510",
     "PHYS 1050",
     "MATH 1210",
     "ENG 1450",
     "COMP 1012"],
    ["MATH 1710",
     "CHEM 1100",
     "ENG 1440",
     "ENG 1430",
     "Written Req"]
]

input_data_values = {
    "courses_taken" : previous_courses_taken, # course codes by semester
    "max_courses_per_semester" : 5, # 4, 5, or 6
    "desired_remaining_semesters" : 8, # 1, 2, 3, ..., or 14
    "desired_courses_by_semester" : [5,5,5,5,5,4,4,4],
    "utility_weightings" : [AVERAGE_BAL, # U1
                           AVERAGE_BAL, # U2
                           AVERAGE_BAL], # U3
    "cost_weightings" : [MORE_BAL, # C1
                        MORE_BAL, # C2
                        MORE_BAL], # C3
    "interest_weightings" : {
        "Engineering Design Process" : AVERAGE_BAL, # I1
        "Mathematics" : AVERAGE_BAL, # I2
        "Program Design and Data Structures" : AVERAGE_BAL, # I3
        "Digital Systems Design" : AVERAGE_BAL, # I4
        "Computer Architecture" : AVERAGE_BAL, # I5
        "Electronics" : AVERAGE_BAL, # I6
        "Software Engineering" : AVERAGE_BAL, # I7
        "Mechatronic Design" : AVERAGE_BAL, # I8
        "Computer Communications" : AVERAGE_BAL, # I9
        "Distributed Systems" : AVERAGE_BAL, # I10
    },
}

```

Figure 27. Input example with course history and value trade-offs

In this input example, 2 semesters of courses had been taken, 8 more semesters were expected until graduation, 5 was the expected enrollment per semester, cost was selected to be “more”

maintained while the other value aspects were selected to be “average” maintained. The resulting pathway recommendation is shown below in Figure 28.

```
[ 'COMP 1012', 'ENG 1450', 'MATH 1210', 'MATH 1510', 'PHYS 1050' ]
[ 'CHEM 1100', 'ENG 1430', 'ENG 1440', 'MATH 1710', 'Written Req' ]
[ 'CHEM 1122', 'ECE 2220', 'ENG 1460', 'ENG 2040', 'MGMT Complement Elec 2' ]
[ 'ANTH 2430', 'ECE 2262', 'ECE 3610', 'ENG 3000', 'MATH 2132' ]
[ 'ARTS Complement Elec 2', 'COMP 1020', 'MATH 2130', 'OTHER Sci Elec 2', 'PHYS 2152' ]
[ 'CHEM PHYS Sci Elec 1', 'COMP 2140', 'COMP Tech Elec 1', 'ECE 2160', 'MATH 2136' ]
[ 'COMP Tech Elec 2', 'ECE 3740', 'ECE 4240', 'ECE Tech Elec 1', 'STAT 2220' ]
[ 'ECE 2400', 'ECE 3700', 'ECE 3760', 'ECE 3780' ]
[ 'COMP 3010', 'ECE 3400', 'ECE 4600', 'ECE Tech Elec 3' ]
[ 'COMP Tech Elec 5', 'ECE 4150', 'ECE 4600b', 'ECE 4830' ]
```

Figure 28. Complete course pathway recommended from the input example

The recommended course pathway in Figure 28 met all graduation requirements and was completed in the expected 8 semesters given the 2 semesters already completed. To demonstrate that sampling the output distribution of the machine learning model can result in different recommended course pathways, Figure 29 presents another recommendation using the same input.

```

['COMP 1012', 'ENG 1450', 'MATH 1210', 'MATH 1510', 'PHYS 1050']
['CHEM 1100', 'ENG 1430', 'ENG 1440', 'MATH 1710', 'Written Req']
['ECE 2220', 'ENG 2040', 'ENG 3000', 'MATH 2130', 'OTHER Sci Elec 2']
['CHEM 1122', 'COMP 1020', 'ECE 3610', 'MATH 2132', 'MGMT Complement Elec 1']
['COMP 2140', 'ECE 2262', 'ENG 1460', 'PHYS 2152', 'STAT 2220']
['ANTH 2430', 'ECE 2160', 'ECE 2400', 'ECE 3700', 'MATH 2136']
['COMP 3430', 'ECE 3740', 'ECE 3780', 'ECE 4240', 'OTHER Sci Elec 1']
['COMP Tech Elec 3', 'ECE 3760', 'ECE 4830', 'ECE Tech Elec 3']
['ECE 3400', 'ECE 4260', 'ECE 4600', 'ECE Tech Elec 1']
['COMP Tech Elec 2', 'COMP Tech Elec 5', 'ECE 4600b', 'MGMT Complement Elec 2']

```

Figure 29. Regenerated complete course pathway recommended from the input example

The regenerated output in Figure 29 is different from the original output in Figure 28 but maintained the same length and number of courses in each semester. The maintained values in each output is also different from the original output and does not always match the input maintained value trade-offs as shown below in Table 29.

Table 29. Maintained value trade-offs input and output

Value	Input	Output 1	Output 2
Utility1	Average	Average	More
Utility2	Average	Average	Average
Utility3	Average	More	More
Cost1	More	More	More
Cost2	More	More	More
Cost3	More	Average	Average
Interest1	Average	Average	Average
Interest2	Average	Average	Average
Interest3	Average	More	Average
Interest4	Average	Less	Average
Interest5	Average	Less	Average
Interest6	Average	Average	Average
Interest7	Average	Average	Less
Interest8	Average	Average	Average
Interest9	Average	Less	Average
Interest10	Average	Average	Average

Output 1 (from Figure 28) has a maintained value accuracy of 0.63, and the regenerated Output 2 (from Figure 29) has a maintained value accuracy of 0.75. This demonstrates a potential for variance in the maintained value trade-offs despite using a highly accurate machine learning model. To facilitate a more holistic evaluation of my CRT, beyond the underlying machine learning model, in the next chapter I present my mixed methods evaluation of my CRT, in which I investigated my CRT as a black box.

Chapter 6: Evaluation

The final phase of my project was the evaluation of my course recommendation tool (CRT), highlighted in Figure 30.

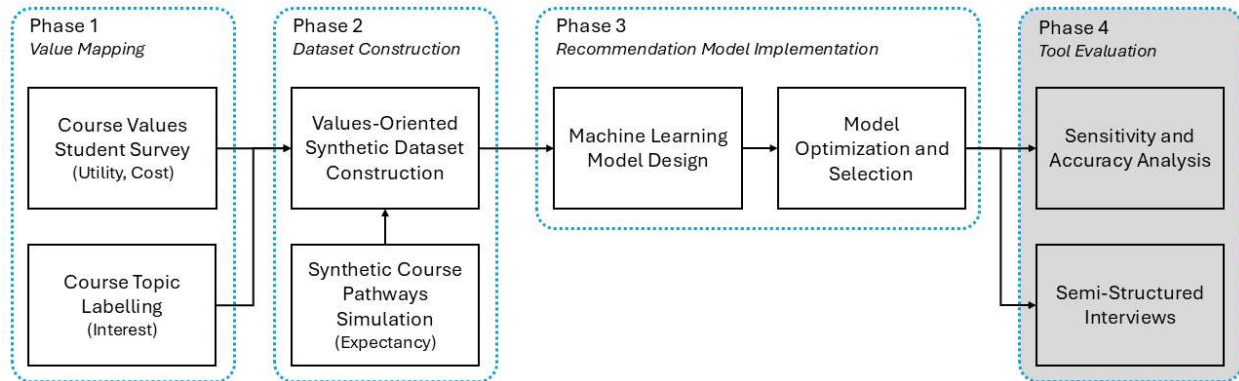


Figure 30. Phase 4: CRT Evaluation

I evaluated my CRT through a mixed methods design drawing from the Use Branch of the evaluation theory tree (Alkin 2004; Mertens 2018). I viewed the feasibility of my CRT through its usability, which aligned with the Use Branch and facilitated my use of methods inspired by value sensitive design (VSD).

To perform this evaluation, I drew from evaluation techniques in the research areas of VSD and recommender systems, and the research topic of human-centered algorithm design, which all call for participatory evaluations situated within the real-world context of the evaluand (what was being evaluated). For my evaluation, I conceptualized my CRT as a black box meaning I only focused on inputs (maintained value trade-offs choices) and outputs (course pathways) of the

CRT. I used two quantitative approaches followed by one qualitative approach to evaluate the tool through the lens of its design purpose and to better understand how well the tool functions.

The sensitivity analyses targeted two aspects of sensitivity; how sensitive each output is to a change in the input and how the range of inputs effects the output. For the first sensitivity analyses, I investigated the effectiveness of each value-oriented input to produce its corresponding value-oriented output in the generated course pathway. For example, how well the three utility inputs produce corresponding utility-maintained pathways. By analyzing the individual input sensitivity, I was able to better understand the usefulness of each input parameter. The second sensitivity analysis I conducted focused on how different shapes of input distributions (how many inputs are set to “more”, “average”, or “less” maintained value). By analyzing the inputs as a whole I was able to better understand the limitations of the tool, such as if my CRT was more accurate for inputs with many values set to “more” maintained value or, instead, more accurate when few inputs were set to “more” maintained value. Understanding the effects of input distributions aided in my understanding of which input conditions are more useable than others.

After the sensitivity analyses, I analyzed the accuracy of my CRT for generating course pathway recommendations that match the selected input value trade-offs. I investigated how often each choice of “more”, “average”, or “less” maintained value was met in the output pathway under the same input distributions for the sensitivity analysis. By investigating the accuracy of my CRT, I better understood how well it met the design goal of generating values-oriented course pathways and how much resampling (rerunning recommendations) would be needed to make accurate recommendations. Accuracy is a key aspect of usability and although resampling can support low

accuracy rates, if the accuracy is too low it may take too much time to generate a values-oriented course pathway, which may lower the perceived usability of my CRT for users.

Following the quantitative analyses, I conducted a qualitative user study to gather the perspectives of computer engineering students (the intended users of my CRT) to aid in the evaluation and my understanding of how my CRT is perceived. The aim of my study was to evaluate how useful my CRT was for computer engineering students, which could not be understood through only quantitative data. The protocol for my interview was inspired by VSD's methods of value-oriented semi-structured interviews and value-oriented prototype (Friedman and Hendry 2019). As such, my interviews were semi-structured and involved using my CRT as a prototype to elicit feedback about students' course choice values and perceptions of my CRT. By investigating the usefulness of the tool through the perceptions of students, this evaluation aligned itself with the participatory evaluation recommendations of human-centered algorithm design (HCAD) (Bauer et al. 2024; Baumer 2017), and VSD (Friedman and Hendry 2019).

Conducting an evaluation, as described as part of evaluation theory (Shadish 1998), consists of four parts: selecting criteria, setting performance levels for the criteria, gathering performance data, and integrating results into a value judgement. However, the exploratory aspects of my evaluation did need criteria and performance levels. As such, the first two steps of evaluation (selecting criteria and setting performance levels) are outlined for value input sensitivity and accuracy criteria in the following sections. The input distribution sensitivity is explorative, so evaluation criteria was not needed. Similarly, the qualitative portion of the evaluation focused on exploring the perceptions of the interview participants.

6.1. Sensitivity Analysis

The first quantitative evaluation approach was based in sensitivity analysis to view the effectiveness of generating values-oriented course pathways that supported maintained value trade-offs selected in the input. Since the primary goal of my CRT was to generate course pathways for computer engineering students that maintained their value trade-offs, the inputs for each of the 16 value aspects had 3 categories of “more”, “average”, and “less” maintained value to choose from. I used the course pathways recommended by my CRT to compute how accurately the inputted value trade-offs were maintained in the outputted course pathway recommendation. As such, sensitivity analysis enabled me to observe which input choices were more influential than others on the accuracy of maintaining inputted value trade-offs.

Within sensitivity analysis methodology review by Borgonovo and Plischke (2016), there are broadly two categories of local and global analysis. Local sensitivity methods focus on a particular point (a single input) in the input space of a model, whereas global sensitivity methods are concerned with the whole input space (Borgonovo and Plischke 2016). For my evaluation, I used two global methods and one local method. The global methods consist of one variance-based approach (Sobol indices) and one density-based (Delta moment-independent indices) approach. Variance-based approaches are concerned with the change seen in the output when an input is changed, whereas density-based approaches analyze the whole output probability distribution when an input is held constant (Borgonovo and Plischke 2016).

Both global sensitivity methods I applied relied on computing the sensitivity of input parameters to a single scalar metric found in the output. Given that inputted maintained value choices were three categories referring to “more”, “average”, and “less” maintained value for each value aspect, I categorized the maintained values in the outputted course pathway recommendations

using the same categories and compared the input against the output. As such, I defined a mean binary accuracy metric (either correct or incorrect) for each of the input parameters. The input parameters for my CRT are displayed in Table 30.

Table 30. CRT input parameters

Input Parameter	Input Choices	Represented Meaning
Expectancy1	4, 5, 6	Maximum number of courses to enroll in each semester based on desired graduation semester
Utility1	More, Avg., Less	Course usefulness to the program
Utility2	More, Avg., Less	Course importance to the program
Utility3	More, Avg., Less	Course usefulness to careers
Cost1	More, Avg., Less	Gave up other activities for the course
Cost2	More, Avg., Less	Needed more time for the course
Cost3	More, Avg., Less	Course lessons were too difficult
Interest1	More, Avg., Less	Engineering Design Process
Interest2	More, Avg., Less	Mathematics
Interest3	More, Avg., Less	Program Design and Data Structures
Interest4	More, Avg., Less	Digital Systems Design
Interest5	More, Avg., Less	Computer Architecture
Interest6	More, Avg., Less	Electronics
Interest7	More, Avg., Less	Software Engineering
Interest8	More, Avg., Less	Mechatronic Design
Interest9	More, Avg., Less	Computer Communications
Interest10	More, Avg., Less	Distributed Systems

Expectancy for this analysis had 1 parameter while the values of utility, cost, and interest values had three, three, and ten parameters respectively. By grouping the input parameters of each value, I defined single scalar metrics for sensitivity the analyses, as the mean binary accuracy of each value. I derived expectancy accuracy based on the ceiling of the fastest graduation time given the maximum number of courses to take each semester, e.g. 5 maximum courses each semester with 27 courses remaining should take 6 semesters to complete.

6.1.1. Sobol Indices

The first global sensitivity method was Sobol indices. Computing Sobol indices (Sobol 2001) is a common variance-based global sensitivity analysis method (Borgonovo and Plischke 2016) that uses Monte Carlo estimations (randomness) to compute first (individual), second (pairings), and total order sensitivity. As a variance-based method, Sobol analysis estimates the fraction of variance in the output metric that is contributed by each input parameter alone, in pairs, and in combination with all input parameters. My analysis used the first order indices to focus on the direct impact of each individual input variable on the output variance. To perform the Sobol analysis, I used the SALib open source python library (Herman and Usher 2017), which implements the works of (Saltelli 2002; Saltelli et al. 2010; Sobol 2001) for Sobol analysis. The Sobol indices were generated using $N=1024$ and a bootstrapped confidence interval of 95%.

6.1.2. Delta Moment-Independent Indices

The second global sensitivity method I used was delta moment-independent indices, which is also known as Borgonovo delta analysis. As a density-based method, Borgonovo delta analysis

estimates how much each input parameter affects the probability distribution of the output metric (Borgonovo 2007). Borgonovo recommended using this method alongside variance-based methods for more complete insights of input interactions, which I followed in this evaluation. To perform the Borgonovo delta analysis, I used the SALib open source python library (Herman and Usher 2017), which implements the works of (Borgonovo 2007; Plischke et al. 2013) for delta moment-independent analysis. The delta moment-independent indices were generated using $N=2048$ and a bootstrapped confidence interval of 95%.

6.1.3. Input Distribution Analysis

The third sensitivity approach I used, and the only local sensitivity approach, was an input distribution analysis. Analyzing the output sensitivity to various input distributions is a common approach for investigating models (Adomavicius and Zhang 2012; Ju 2009; Lessmann et al. 2005; Petrik et al. 2013; Singh et al. 2021). Through this analysis, I investigated how the accuracy of my CRT varies when different EVT input choices were made to simulate different value trade-offs. For example, one user might want most of the values maintained, forming an input distribution with mostly “more” inputs, whereas another user might only care to maintain a particular value, which would form an input with one “more” and the rest may be “average” or “less”. Although I designed my CRT to recommend maintained value trade-offs within course pathways, a user may want a value to be “less” maintained. As such, this analysis revealed if certain trade-offs are necessary between the amount of maintained value for each input parameter. Examples of these potential trade-offs could be higher accuracy in cases of equal portions of “more” and “less”, or an inverse relationship of a decreasing in accuracy when increasing the number of inputs set to “more”. During each distribution test, each input

parameter was randomly sampled from the same input distribution following the probability distribution approximations in Table 31.

Table 31. Input distributions for accuracy analysis

Input Distribution	Input Maintained Value Probabilities		
	More	Average	Less
Uniform	0.333	0.333	0.333
Normal	0.159	0.682	0.159
Bimodal	0.425	0.150	0.425
Left Triangle	0.500	0.333	0.166
Right Triangle	0.166	0.333	0.500

I analyzed five common input distributions: uniform, normal, bimodal, right triangle, and left triangle alongside the conditions of five input pathway lengths representing zero, two, four, six, and eight semesters of courses completed. Eight semesters was included as it is common for engineering students from this particular program graduate in more than four years (recall from Chapter 4 that summer semesters are not modelled in the tool, so each year has 2 semesters). Each probability distribution was investigated with each of the input pathway lengths using $N = 1000$ synthetically simulated course pathways. That is, 5000 pathways separately recommended for each of the 5 input distributions.

6.1.4. Value Sensitivity Criteria

The ideal sensitivity results regarding usage of my CRT for each metric described above were that the outputs associated with each EVT construct (expectancy, utility, cost, and interest) have the highest sensitivity to their respective input parameters. As the values of utility, cost, and interest are intertwined within each course, certain input parameters may not be as influential as others. I presented the results of the Sobol and Borgonovo delta indices as rankings of sensitivity, which is more meaningful than the actual number for comparing usefulness in this evaluation. By ranking input sensitivities (influence), the ideal case is when the value parameters are the highest ranked for their value metric, and conversely the least favorable case is when the value parameters are the lowest ranked for their respective metric. I defined the performance criteria between these ends of the spectrum.

Defining rank-based performance levels for value sensitivity must vary between values since, with the exception of utility and cost, each value has an unequal number of input parameters. To account for this, I defined four performance levels as a range of rankings based around the ideal case for each value. Table 32 displays the performance level ranges (inclusive) of each value.

Table 32. Performance levels for value sensitivity

Performance Level	Sensitivity Rankings of Each EVT Construct			
	Expectancy	Utility	Cost	Interest
Ideal	1	1 – 3	1 – 3	1 – 10
Good	2 – 3	4 – 5	4 – 5	-
Adequate	4 – 6	6 – 7	6 – 7	11 – 12
Unsatisfactory	7 – 17	8 – 17	8 – 17	13 – 17

This approach allowed me to discuss parameters separately and as a collective of contributions to the usability of the CRT. As interest parameters encompass 10 of the 17 input parameters, I removed the performance level “Good”, which resides as the level below “Ideal”, since a sensitivity ranking of 11 (one rank outside of ideal) cannot be referred to as having “Good” sensitivity. These performance levels aided in identifying how well the input choices are reflected in the recommended course pathways and provided insights into whether any parameters were not effective.

6.2. Pathway Value Accuracy Analysis

In addition to assessing the sensitivity of my CRT, I analyzed the overall accuracy of the generated course pathways to situate my evaluation within the context of how my CRT was intended to be used. The data collection for this analysis of accuracy took place alongside the data collection for the input distribution analysis. As such, I conducted tests for each of the five input distributions paired with each of the five input pathway lengths consisting of measuring the accuracy across 1000 simulated course pathways. This approach led to 25 accuracy results for each of the pairings. I computed accuracy as a binary result of whether or not the intended maintained value category sampled for the input (“more”, “average”, or “less” maintained value) was the same category in the generated course pathway. I computed the expectancy accuracy based on whether or not the estimated remaining semesters to graduate was met and how many semesters predictions were off by when inaccurate.

6.2.1. Value Accuracy Criteria

For the CRT to be useable, it is needed to be accurate. From the model selection results in Chapter 5, the best model integrated into the CRT had an accuracy (recall@6) of 0.9. That accuracy is for predicting the next semester of courses given EVT parameter inputs and a transcript of completed courses. The accuracy of the ML model did not investigate the accuracy of generated course pathway recommendations. Since the model was integrated into the CRT, the model can be resampled (re-run) if the generated course pathway recommendation is not sufficiently accurate. Resampling is a benefit of being able to measure accuracy immediately after generation. However, resampling was not used in the quantitative evaluations. It is important to note that total accuracy for each input parameter is not probable since value trade-offs are a natural aspect of course select and the more courses are completed the less freedom there is to generate values-oriented pathways within the desired graduation time. However, if the accuracy is too low, the tool cannot reasonably regenerate course pathways in a timely manner, which may negatively impact the usability of the tool. Table 33 outlines the performance levels for input accuracy.

Table 33. Performance levels for input accuracy

Performance Level	Input Accuracy (<i>acc</i>)
High	$acc \geq 0.7$
Moderate	$0.7 > acc \geq 0.3$
Low	$0.3 > acc$

Broad accuracy bins of “high”, “moderate”, and “low” were used to categorize the input accuracy of each of the 25 tests. To interpret the accuracy results, I chose to discuss how likely

each level of accuracy is to appear in the output based on their proportion of all 25 tests as outlined in Table 34.

Table 34. Performance levels for likelihood of accuracy

Performance Level	Likelihood of accuracy (L)
Very	$L \geq 0.7$
Somewhat	$0.7 > L \geq 0.3$
Not very	$0.3 > L \geq 0.1$
Not at all	$0.1 > L$

These performance levels for likelihood from very likely to not at all likely were used to broadly evaluate the likelihood of each performance level for accuracy. For example, a value may have had likelihood results of 0.25, 0.7, 0.05 for the performance levels of “high”, “moderate”, and “low” respectively. As such, this case would be not very likely to have high performance, very likely to have moderate performance, and not at all likely to have low performance. I also set performance levels for the number of semesters the generated pathway was over the estimation when inaccurate, outlined in Table 35.

Table 35. Performance levels of extra semesters when inaccurate expectancy

Performance Level	Extra Semesters when Inaccurate ($extra$)
High	$extra < 1.5$
Moderate	$1.5 \leq extra < 2$
Low	$2 \leq extra$

As an inaccurate expectancy already results in at least one more semester, I chose to evaluate based on being close to only being one semester over. These performance levels aided in my interpretation of the value accuracy results.

6.3. Semi-Structured User Interviews

To further situate my evaluation within the context of using my CRT, I conducted semi-structured interviews to understand current computer engineering students' perceptions of my CRT and its recommended course pathways. Two participants were recruited from the same pool of students who participated in the value-mapping survey (Chapter 3) in Phase 1. Both participants had indicated their interest in participating in future phases of the research. To be eligible for the interviews, participants must 1) have been current computer engineering students at the University of Manitoba who, 2) had completed at least 60 credit hours of courses within the program, 3) had started the program in a Fall semester, and 4) expected to have three or more semesters remaining before graduation. These inclusion criteria ensured that my CRT's recommendations could be used by participants and that participants had prior experiences choosing courses to reflect on.

I designed the interview protocol similar to value-oriented techniques for semi-structured and prototype interviews described by VSD (Friedman and Hendry 2019). As such, my interview protocol focused on exploring participants' course choice values and examining their evaluations of my CRT after using my CRT prototype. I conducted the semi-structured interviews online for approximately one hour each. The interviews allowed for coverage of key questions while also allowing for probing students to further understand their perspectives. During the interviews,

participants used my CRT through myself as a proxy (I inputted their choices and displayed the recommendations); reflected on at least three different pathway recommendations; and provided feedback on their impressions about my CRT. Since my CRT is probabilistic and pathway length is an important aspect of course planning, I regenerated pathway recommendations (through a Python script) until a pathway matching the length of the participant's desired pathway length was generated within 100 attempts, otherwise returning the shortest length pathway recommendation if the limit was reached. This number of attempts allowed sufficient room generating a valid pathway within 30 seconds based on my computer hardware running the underlying ML model (Nvidia RTX 3060 Ti).

My analysis of the interview transcripts focused on the current course planning supports used by participants, the impact values have on their course planning, their perceptions of my CRT and their suggestions for improving my CRT. To analyze the interview transcripts, I applied two types of content analysis. First, I conducted a conventional content analysis (inductive approach) (Hsieh and Shannon 2005) to gain an understanding of the current supports used by participants during their course planning process. By better understanding current supports, I aimed to improve my understanding of the context that my CRT was being added to. Next, I used a directed content analysis (Hsieh and Shannon 2005; Zhang and Wildemuth 2009) with EVT being the theory providing direction for coding. Since EVT was the framework for which student values were integrated into the tool, evaluating the tool with respect to the same value framework allowed for better comparison and further directions to take the tool based on the results.

While analyzing the results of the directed content analysis for participant perceptions of my CRT and suggestions for improvement, I noticed that the elicited values were regarding motivation for two different tasks, 1) course planning and 2) using my CRT. As EVT connects a

person's expectations about their success to perform a task with their values about the task, distinguishing between the tasks was important. That is, some perceptions and suggestions were regarding their motivation for choosing their courses, while others were focused on their motivation for using my CRT. As such, I separated my analysis of perceptions and suggestions into the motivations guiding each of the two tasks.

6.4. Evaluation Results

Results from my evaluation showed ideal input sensitivity for input parameters for each EVT construct from the Sobol analysis and near ideal sensitivity from the Borgonovo delta analysis. The accuracy of the tool was found to be agnostic to the shape of the input distributions and varied as the number of semesters completed in the input increased. To present course pathway recommendations with high accuracy for maintained value trade-offs and length, I found that resampling was required. To achieve high performance, much resampling was needed whereas moderate performance was achieved with few attempts.

Based on the current course planning supports available to students, my CRT can aid students and potentially academic advisors. I found that expectancy, attainment, cost, and interest impacted participants' course planning. Overall, participants perceived my CRT as useful for presenting what courses they have remaining in their program and what order courses can be completed to graduate on time. Suggested additions to my CRT centered around supporting participants' interest and utility.

6.4.1. Sensitivity Results

The results from the first order Sobol indices and the Borgonovo delta moment-independent indices are shown as average ranks in Table 36 where 1 is the most sensitive input parameter and 17 is the least sensitive.

Table 36. Average parameter sensitivity ranks under each EVT construct accuracy metric (lower is more sensitive; with ties)

Input Parameter	Sobol Rank				Delta Rank			
	E	U	C	I	E	U	C	I
Expectancy1	1	5	11	14	1	10	14	14
Utility1	4	1	4	15	5	2	12	13
Utility2	3	2	5	16	2	4	8	11
Utility3	4	3	5	17	16	1	3	17
Cost1	7	16	1	13	17	15	1	12
Cost2	12	17	3	12	7	4	4	14
Cost3	16	9	2	11	9	7	1	16
Interest1	8	4	14	9	11	13	4	10
Interest2	9	12	17	10	4	12	16	9
Interest3	2	6	13	8	13	17	10	5
Interest4	6	14	8	6	12	13	8	6
Interest5	16	9	10	1	14	16	4	7
Interest6	15	13	8	4	15	3	11	7
Interest7	11	7	15	2	6	4	13	2
Interest8	9	15	12	6	7	8	15	3
Interest9	13	8	7	4	10	11	17	4
Interest10	14	11	15	3	3	8	7	1

These averages were across tests that generated course pathways from zero, two, four, six, and eight semesters of completed courses. Only the generated portion of the course pathway was analyzed for categories of maintained values. The highlighted sections within Table 36 are the ranks of the parameters intended to affect the value accuracy metric for that column.

The single expectancy input parameter was rank 1 for both analyses when focused on expectancy accuracy. The three utility input parameters were the top three ranks from Sobol and ranks 1, 2, and 4 from Delta when focused on utility accuracy. The three cost input parameters were the top three ranks from Sobol and ranks 1, 1 (tied for first), and 4 from Delta when focused on cost accuracy. Finally, the ten interest input parameters were ranks 1 through 10 for both Sobol and Delta analysis when focused on interest accuracy.

The input parameters associated with their respective EVT constructs were able to successfully be ideal factors influencing the levels of maintained values in the generated course pathways from various stages in the computer engineering program. However, two cases occurred where the influence of an input parameter was one rank below the range of their ideal rank. These two cases were from the Delta analysis and the values of utility and cost. For utility sensitivity, the “utility2” parameter was ranked 4th (tied with “cost2” and interest7”) whereas “interest6” was ranked 3rd. For cost sensitivity, “cost2” was ranked 4th (tied with “interest1” and “interest5”) whereas “utility3” was ranked 3rd. Parameters for the utility and cost analyses are considered to have good sensitivity when ranked 4th. See Appendix F for plots of the change in parameter ranking at each amount of input semesters and proportional comparison plots of average output variance of each parameter. Next, I present the sensitivity of input distributions on the accuracy of my CRT.

The results from the input distribution sensitivity analysis were derived from the binary maintained value accuracy for each value, as a mean of the 16 maintained value input parameters (3 utility, 3 cost, and 10 interest parameters). The mean maintained value accuracy across each of the five input distributions is presented in Table 37.

Table 37. Mean binary accuracy for each maintained value bin across values (weighted equally) by distribution (higher is more accurate)

Input Distribution	Mean Accuracy		
	More	Average	Less
Uniform	0.212	0.636	0.199
Normal	0.208	0.636	0.215
Bimodal	0.213	0.621	0.200
Left Triangle	0.218	0.627	0.200
Right Triangle	0.206	0.629	0.214

There is little change in the maintained value accuracy of all value inputs for each input distribution. This suggests that the input distribution for my CRT had little effect on the accuracy of maintaining value trade-offs in generated course pathways. However, I observed the maintained value accuracy across changes based on the length of the course history, in Table 38.

Table 38. Mean binary accuracy for each maintained value bin across values by semesters

Input Semesters	Mean Accuracy		
	More	Average	Less
0	0.040	0.556	0.442
2	0.121	0.747	0.183
4	0.174	0.742	0.146
6	0.232	0.696	0.128
8	0.490	0.408	0.129

“More” maintained value was very difficult to obtain from zero input semesters and was easier to obtain from 8 semesters. “Average” maintained value became more accurate in the middle of the program and less accurate towards the end. “Less” maintained value was often accurate when no courses had been completed and was somewhat difficult to obtain after courses had been completed with later years having the lowest accuracies. As these results were meant to explore averages across all 16 value inputs, these results did not capture the variations in test accuracy.

The results from the Sobol and Borgonovo analyses demonstrated that the CRT input decisions users make regarding their course selection motivational values were directly connected to the CRT course pathway recommendations. As the rankings were averages across recommendations generated from pathways of varying numbers of input semesters, the input values correctly influenced the output values regardless of how many courses a user had completed. The analysis of the accuracy of my CRT under different input distributions showed little variation between distributions and suggested that my CRT was agnostic to the shape of the input. This means that any configuration of input value trade-offs is likely to be as accurate as another. In addition, I found that my CRT was sensitive to the number of input semesters meaning that accuracy will be different for students at different stages in the curriculum. Next, I discuss the accuracy of my CRT to better understand the variance in maintained value accuracy and how much resampling may be needed for high performance.

6.4.2. Pathway Maintained Values Accuracy Results

Although I presented the accuracy of my CRT in the input distribution analysis, averaging accuracy across each value was not representative of the individual accuracies of each value. As

such, to better represent the accuracy of my CRT, I investigated each of the 25 accuracy tests (five input distributions tested with five numbers of input semesters) for each input parameter to see the proportion of tests with high, moderate, or low accuracy performance. To aid in interpreting the accuracy results, I present the likelihood of accuracy performance levels for values and maintained value categories in Table 39.

Table 39. Maintained accuracy performance likelihood results

Accuracy	Utility	Cost	Interest	More	Average	Less	All
High	0.222	0.240	0.120	0.010	0.373	0.103	0.162
Moderate	0.209	0.173	0.368	0.220	0.473	0.213	0.302
Low	0.569	0.587	0.512	0.770	0.155	0.685	0.537

Utility and cost were not very likely to have high or moderate accuracy and were somewhat likely to have low accuracy. Interest was not very likely to have high accuracy and was somewhat likely to have moderate or low accuracy. These results suggested that resampling is required to recommend accurately maintained values across course pathways to users since low accuracy was the most likely. To generate high accuracy pathways, five to nine generations may be required to achieve high accuracy for each of the three EVT values.

In term of maintained value categories, “more” was not at all likely to have high accuracy; was not very likely to have moderate accuracy; and was very likely to have low accuracy. “Average” was somewhat likely to have high or moderate accuracy and was not very likely to have low accuracy. “Less” was not very likely to have high or moderate accuracy and was somewhat likely to have low accuracy. Across each of the three EVT values, my CRT was not very likely to have high accuracy and was somewhat likely to have moderate or low accuracy. For inputs for “more”

maintained value to have high accuracy recommendations, a mean of approximately 100 pathway generations is expected. However, in 5 pathways generations it is expected that moderately accurate recommendations will be achieved for “more” and “less” maintained values while also achieving high accuracy for “average” maintained values.

Although these results initially appeared to suggest that my CRT had low performance, due to the maintained value modelling assuming a normal distribution, inaccuracies have different meanings. When “more” or “less” maintained value was inaccurate, I found the results were “average”. However, when selecting “average” maintained value, inaccuracies resulted in both cases of “more” and “less”. Table 40 presents the likelihood of increasing maintained values from “average” to “more” and categorized by the mean probability of improvement across each of the 25 tests.

Table 40. Likelihood of maintained value improvement from “average” to “more” when inaccurate for “average”

Maintained Value Improvement Probability (<i>I</i>)	Utility	Cost	Interest
High ($I \geq 0.7$)	0.600	0.453	0.236
Moderate ($0.7 > I \geq 0.3$)	0.187	0.440	0.264
Low ($0.3 > I$)	0.213	0.107	0.500

Utility was somewhat likely to have a high probability of improvement and was not very likely to have moderate or low probability of improvement. Cost was somewhat likely to have a high or moderate probability of improvement and was not very likely to have a low probability of improvement. Interest was not very likely to have a high or moderate probability of improvement and was somewhat likely to have a low probability of improvement. These results suggested that

course pathways were likely to be generated with “more” maintained value for utility and cost during inaccuracies for “average” whereas interest was more likely to have a reduction in maintained value when inaccurate for “average”. Table 41 presents the expectancy accuracy performance and the performance of the extra semesters when inaccurate.

Table 41. Expectancy accuracy performance likelihood results

Accuracy	Expectancy
High	0.000
Moderate	0.000
Low	1.000
Extra Semesters when Inaccurate	
High	1.000
Moderate	0.000
Low	0.000

Only low performance was achieved for expectancy accuracy with a mean of 0.144 and a standard deviation of 0.08 across all 25 tests. However, the number of extra semesters when inaccurate has high performance with a mean of 1.182 and a standard deviation of 0.05. These results suggested that an average of approximately seven pathway generations may be required to find a pathway that completes the program within the expected amount of time. When pathways were generated and the desired completion time is not met, it was very unlikely to be a full year (two semesters), or more, beyond the expected completion and was very likely to be one semester beyond.

As the overarching goal of my CRT was to recommend course pathways with maintained values, it was important to understand the effects of categorical maintained value inaccuracies to better evaluate performance and to guide usage of my CRT and design improvements. From the

maintained value accuracy results, the maximum expected number of generated pathways to discover one pathway with high performance was 9 whereas 100 generations were expected for high performance for “more” maintained values. This suggested that 100 course pathway recommendations should be generated for each user request where each recommendation is analyzed for accuracy before the most accurate recommendation is presented to the user. When analyzing inaccuracies, I found that it was likely for inputs set for “average” maintained values to result in “more” maintained value when inaccurate, which suggested that accuracy based on comparing the input and the output maintained value categories on a binary basis may have been insufficient. This insufficiency brought to light a potential design change for my CRT where user inputs for “more” maintained value would be prioritized in accuracy calculations. For expectancy, the low accuracy requires approximately 7 pathway generations to be accurate for pathway length although the majority will be one semester longer than expected. Since the maximum average number of regenerations for high performance was 100, my CRT should generate at least 100 pathways recommendations and analyze the accuracy of each recommendation before presenting the most accurate values-oriented course pathway to the user. Alternatively, for at least moderate performance, 7 pathways should be generated with expectancy accuracy as the limiting factor.

6.4.3. Interview Findings

I separated the interview findings into four parts to explore students’ contexts surrounding course planning; to explore how values impact students’ course selection experiences; to understand students’ perceptions of my CRT for evaluation; and to present students’ suggestions of how to improve my CRT. First, I discuss the course planning supports mentioned by participants to

understand what supports are used and how my CRT can further contribute to available supports. Second, I discuss the perceptions of how values impact participants' course planning through the lens of EVT by connecting back to the values integrated into my CRT. By connecting students' perceptions back to EVT I furthered my understanding of how values affect course planning. Third, I present participants' perceptions of my CRT through the EVT framework in the two cases of the task being motivated as 1) choosing courses and 2) when the task is using my CRT, which both aided in the evaluation of my CRT. Finally, I present suggestions made by the participants to improve my CRT, which was also analyzed through the EVT framework in the cases of the task being motivated as 1) choosing courses and 2) when the task is using my CRT. The distinction of motivation for different tasks was a key part in the analysis of the transcripts. For example, if my CRT was perceived as having high utility, it would be important to distinguish between the meanings of my CRT supporting utility-focused course planning, or my CRT being useful for students. By understanding the motivations behind how my CRT can support course planning, I could better evaluate my CRT and gained insights on future CRT design directions, whereas by understanding what motivates using my CRT I could further recommend future CRT design directions focused on the user experience.

6.4.3.1. Supports for planning courses

From the inductive content analysis of the 2 interview transcripts, I found that the primary support for both participants' planning of their course pathway was the Computer Engineering Course Flowchart, as seen in Appendix A. This flowchart is updated each year by the ECE department and visually communicates course prerequisites and the requirements needed to graduate. Other supports used by participants were UM Achieve and the ECE academic advisor,

each mentioned by one of the two participants. UM Achieve enables students to view their grades what courses they require to graduate but does not present course prerequisites or a course pathway. My CRT could be used alongside UM Achieve to aid students in discovering pathways to graduation. The ECE advisor was mentioned by one participant as a very important support, although second to the course flowchart, often meeting a couple times each year for planning assistance. My CRT could potentially be used by academic advisors to help guide students' course planning and it could be useful to show students the limitations to pathway planning such that they know when an override from an advisor could help them achieve their planning goals. One participant also mentioned that a friend of theirs in the mechanical engineering program does not have a course flowchart and has previously expressed jealousy of the ECE flowcharts. This remark suggests that a course flowchart or a similar CRT could be helpful for mechanical engineering students at the University of Manitoba as an additional support.

6.4.3.2. How values impact course planning

I used EVT to categorize the results from the detective directed content analysis into aspects of course choice and planning according to the EVT construct of expectancy and values of attainment, interest, utility, and cost. In these findings, I divided expectancy into two separate categories, one focusing on the expectation to succeed in their chosen courses and the other focusing on their expectations to succeed in planning. In terms of the participants' expectations to do well in chosen courses, both participants believed they could graduate on time given their current approaches to planning. One participant also mentioned that there are variations in course workload with the example that math courses had a relatively bigger workload whereas electives were smaller. For expectations regarding successful planning of courses, both participants rely on

their supports to be accurate. One participant also mentioned that they rely on having other course options when they are unable to follow their planned pathway, while the other participant mentioned their reliance on taking summer courses to graduate faster and planning around when courses were offered in previous years.

One participant expressed their focus on attainment, which was reflected by them planning their course pathway in order to be eligible for bursaries, “being a full-time student's very important to me, [otherwise] I wouldn't be eligible for different bursaries that I get every year” (P2). Although all engineering students are generally required to be full-time, this is not the case in their final year before graduation, this student, therefore, put additional time into planning their schedule to ensure they retained full-time status through their final year. Interest was a priority for both participants and both related interest to their choice of electives. However, the two students expressed this in different ways. One participant chose electives that would allow them to explore new interests and discover what they enjoy, while the other participant chose electives based on topics they already knew they enjoyed. One participant also mentioned that when a course is not of interest to them, but is required, they are sometimes “bummed out” (P2) to register.

Both participants also considered utility when planning their courses. Both participants planned to take capstone, the final year design course that spans the fall and winter semesters, in their final year and planned around meeting all prerequisites for capstone before their planned final year. As such, they chose courses that helped them complete capstone prerequisites in a timely manner. One participant mentioned that they plan to take summer courses each year and since summer offerings are limited, they avoid taking courses offered in the summer semester during the fall or winter semesters. The other participant accounted for some courses that are offered

only once a year but are a critical prerequisite for other courses. These utility-centered approaches to course planning are also tied to the perceived cost of the time taken to graduate and being able to graduate within their expected timeframe.

Cost appeared to be the most significant value that affected course choices and planning, particularly when it came to balancing the perceived cost of different courses. Both participants plan around graduating “on time”(P1, P2) and balancing remaining courses equally across their remaining semesters. Another aspect of balancing mentioned by one participant was attempting to balance out their course workload within their semesters by taking an elective course to balance-out a math course, “I'd be like, okay, I'm taking Math 2, like, I'm gonna take an elective also” (P2). One participant chose courses that are all on the same days of the week to free up time on the days without any courses,

“whenever I'm applying for, like, courses, [what] I try to do is, if the courses are, like, all in one day. So, like, let's say, let's say, for example, like, my last semester, I took 4 courses. All four of them were Monday, Wednesday, Friday. And I actually like that. like, if all my... most of my courses are on one day, so I can get the rest of the days off” (P1).

The other participant expressed that they spend roughly 30 hours every July planning their courses for the next semester,

“to be honest, I pour over my course selection for, like, probably a cumulative 30 hours every July. Of, like, lining absolutely everything up that I could line up, and just being sure that I want to do certain things” (P2).

This participant also discussed that they felt like they had fewer choices as they become closer to graduation because graduating on time is the highest priority, meaning they may end up with unbalanced semester workloads, or take many courses of low interest.

6.4.3.3. Perceptions of my course recommendation tool

I separated the results from the deductive directed content analysis into the two tasks being motivated. The first task was course planning where I investigated how my CRT supports EVT values. The second task was using the CRT as some responses were focused on participants' usage of my CRT. In both instances, EVT was used to guide my analysis.

Motivating course planning

As a support, both participants' expectancies for choosing courses was supported by my CRT as it was expressed as useful for getting an idea of what courses are needed from their current position in the program up to graduation. One participant was surprised when the recommended pathway better fit their personal balance of expectations and value trade-offs, "I honestly thought I couldn't do it in 4 classes each time. Very cool. And it's much more well-balanced than when I said 5" (P2). Both participants expressed utility of the tool to assist in navigating prerequisites after entering the computer engineering program. Cost associated with choosing courses was also supported as both participants expressed the tool has potential to reduce time and effort of course planning. One participant expressed that an important aspect of my CRT is being able to manage the semester workload through selecting how many courses will be recommended for each semester, "[selecting] how many courses you want to take a term, I think, is the most valuable.

And that... that was cool” (P2). I did not find that my CRT supported attainment, which was not designed into my CRT.

Motivating use of my CRT

Both participants mentioned that course pathway recommendations from my CRT were accurate compared to their current planned pathways, which supports their expectancy of my CRT to be a support. One participant preferred the output from the tool-determined input value profile (based on the maintained value trade-offs in their semesters of completed courses) whereas the other participant preferred the output from the input value profile they created from their own preferences. One participant highlighted that, to them, the most important aspect of the tool was being able to select how many courses will be in each term, meaning this feature increased their expectations for my CRT to support their choices. One participant found interest in the CRT itself and said it was “cool” (P2) at various points throughout the interview. For utility, both participants highlighted that the tool would be most useful to them and others if used after completing the preliminary year courses and entering the program. Both participants also expressed usefulness from the tool displaying the courses needed to graduate from their current position within the program. In terms of cost value, both participants highlighted that the tool had the potential to save time and effort in their planning approaches. There was no mention of perceptions related to attainment.

6.4.3.4. Suggested additions to my course recommendation tool

Similar to participants' perceptions, I separated suggested additions to my CRT, into the two tasks being motivated. The first task was course planning where I investigated how my CRT supports EVT values. The second task was using the CRT as some responses were focused on participants' usage of my CRT. In both instances, EVT was used to guide my analysis.

Motivating course planning

When I asked participants about potential additions or changes they would want to see regarding the course recommendation tool, both participants had a want for recommendations to account for technical concentrations, also known as focus areas, within their program. However, their reasonings for wanting this addition were different. One participant wanted to explore the concentrations and see how they affect their course pathway,

“as computer [engineering students], we can go into software, we can go into embedded hardware, whatever, right? So...I'm still deciding on what, so then I can pick the according technical electives” (P1).

Whereas the other participant wanted more detailed elective recommendations to enable them to complete their biomedical concentration which involves specific courses for natural science and technical electives. Both of these notions are rooted in supporting their interest during course planning. Another addition mentioned by one participant, was to add hyperlinks to each course's information to support reading about the course, see who is teaching it, and register for the course. This idea of hyperlinks to more information could better assist with the participant's expectancy by displaying information to them and support reducing the costs of time and effort

through easier registration. No additions or changes were expressed by participants regarding attainment or utility.

Motivating use of my CRT

Suggestions were also made related to participants' value perceptions of using my CRT. To improve the utility of my CRT, one participant wanted summer semesters to be added to the tool. The other participant stated, "If it took into account the time of day, I would... probably depend on it" (P2). This quote highlights how a utility-focused addition of accounting for the timing of course offerings may affect students' expectancy of my CRT as a support. For improvements to the cost of my CRT, one participant wanted to be able to save recommended course pathways to view later instead of needing to copy it down or save a screenshot. Suggestions for motivating use of my CRT regarding attainment or interest were not expressed by participants.

6.5. Evaluation of my Piloted Values-Oriented Course Recommendation Tool

My evaluation of my pilot course recommendation tool holistically investigated both quantitative and qualitative results. The sensitivity analyses highlight that the tool is useable to generate values-oriented course pathways and is equally useable for any combination of inputs.

Sensitivity showed that each value is sensitive to their respective input parameters, which allows them to properly influence maintained value trade-offs within generated course pathways regardless of the input distributions.

For my CRT to achieve high performance for accuracy of “more” maintained values, an estimated average of 100 course pathways need to be generated whereas at least moderate performance could be achieved in an estimated average of 7 generated pathways. This shows that the tool is useable, however more time (resampling) is required to present highly accurate course pathways with “more” maintained values. Pathways presenting moderate accuracy or presenting “average” maintained value will be found more quickly in comparison. I found that inaccuracies for “more” and “less” maintained value resulting in “average” maintained value whereas inaccuracies for “average” balance tended towards “more” for utility and cost, and “less” for interest. These effects of inaccuracies highlighted that weighting inputs by importance or tracking the number of “more” maintained values in the outputted course pathway may aid in recommending course pathways that better maintain value trade-offs.

It is clear that my CRT can add to the supports available for computer engineering students’ course planning by aiding students and potentially aiding academic advisors. Both of the interview participants indicated that the tool was useful as a support for their expectancy to succeed in their course planning. Cost was expressed by both participants as an aspect of course planning that my CRT has the potential to reduce. Additionally, participants identified aspects of the tool that would further improve the tool’s representation of interest-value through the integration of program concentrations.

Given the results from the quantitative analyses, the tool is able to generate values-oriented course pathways in a useable manner, and the amount of resampling determines the performance of what accuracy is achieved. From the directed content analysis, the participants responded positively to the course recommendation tool, finding it useful as an additional support. As such,

I determined that my CRT had acceptable performance and further research is warranted to refine, and explore changes to, the approach used to design and construct my CRT.

Chapter 7: Discussion

My work integrated the theoretical and participatory approaches of human-centered algorithm design alongside a balance-oriented approach to course choice motivation. Through this framework, I mapped computer engineering students' EVT values onto synthetic course pathways through the curriculum, constructed a dataset of values-oriented course pathways, trained a generative machine learning model on the dataset, and built a prototype course recommendation tool (CRT).

In the first phase of my study, survey results demonstrated that there were ranges of drawbacks and usefulness associated with each course, though most courses were perceived by students as both useful and with low drawbacks. These results were not alarming and presented insights into the value balancing of utility and cost for preliminary year students while also highlighting the complexity of balancing either value across the program. Results from the course topic labelling found that ten subjects (topics of study) within the curriculum occur in at least four courses. As such, ten subjects of interest were able to be modelled for being maintained throughout course pathways. Results also highlighted the prevalence of engineering design, computer programming, and mechatronic design across UM's Computer Engineering program.

When I constructed the dataset of values-oriented pathways in the second study phase, I used the survey results and course topic labelling to formulate a statistical interpretation of maintained value within pathways to differentiate between categories of "more", "average", and "less" maintained value. I used these categories to label the amount of maintained value for the three aspects of utility, three aspects of cost, and ten aspects of interest for each synthetic course pathway. These labels of maintained value represent conditional inputs for the course recommendation tool such that the machine learning model was meant to learn to differentiate

which course pathways contain “more”, “average”, or “less” maintained value for each aspect of utility, cost, and interest.

I then investigated my novel methodology in the third study phase, through designing and implementing a simple Transformer-based machine learning model for course recommendations. I investigated three input representations (CTV, Semester Sequence, and Hybrid) and found that the model performed best when representing the courses completed as a sequence of semesters (similar to a transcript). The best performing model during cross-validation had an average validation accuracy of 0.9 (recall@6) and an average validation ranking ability of 0.97 (AUC), which demonstrated that the values-oriented information in the dataset was effectively learned by the model to a high degree. The model’s ability to generalize the values-oriented information within the dataset demonstrated that my approach is feasible for predicting values-oriented semesters of courses.

Finally, in the fourth phase of the study, I performed a mixed methods evaluation to examine my CRT’s overall feasibility to generate values-oriented course pathways after integrating the machine learning model. The goal of this evaluation was to evaluate beyond the machine learning model’s predictive performance for semesters by analyzing complete course pathway recommendations. I analysed sensitivity and accuracy to validate the usability of many maintained value trade-offs, and I analyzed semi-structured interview transcripts to understand students’ perceptions of my CRT. Sensitivity results showed that the tool’s value inputs effectively influence the balance of values in the generated course pathways, and that the output is agnostic to the shape of the input distribution meaning that any combination of inputs (value orientations) is as accurate as another. Accuracy results showed that the tool requires generating many course pathways and presenting the most accurate pathway to the user. Depending on the

number of pathways generated, it is possible to achieve high performance from many pathways or moderate performance from few pathways. I also found that cases of inaccuracy for “average” maintained value tended to result in “more” maintained value, which showed the tool may have a bias towards “more” maintained value. Based on the accuracy results, my CRT demonstrated high usability through accuracy and the assumption of resampling.

Results from the user study in the fourth phase of my study highlighted the experiences of computer engineering students who briefly used my CRT. Participants expressed that the tool could aid in their expectations of what their course pathway can look like and has the potential to reduce costly aspects of course planning such as time and effort. When using the tool, participants noted that the generated pathways were very similar to their currently planned pathway, which highlighted the tool’s accuracy and extendibility towards real-world application despite being trained on synthetic data. The participants also suggested improvements for the tool that would aid in their course planning experience or in the usability of the tool. From the perceptions of participants, the usability of CRT is further supported and there is potential to add features that further improve its usability and support their course planning values.

Overall, implementation results demonstrated the generative machine learning model’s ability to learn to predict values-oriented semesters of courses constructed from the combination of value mapping, synthetic course pathways, and balance-oriented approach to course selection. The evaluation results demonstrate the tool’s potential to be a highly useful support for engineering students during course planning with further potential for improvement based on participant feedback. From the findings of this research project, I believe my novel method for the development of a course recommendation tool should be feasible to extend to larger sample sizes in engineering programs of similar flexibility.

7.1. Contributions

My contributions are interdisciplinary in nature as this project was a combination of engineering education research and studies of course recommendation systems bridged together by human-centered algorithm design and expectancy-value theory. My research presented in this thesis makes methodological and practical contributions to the research topic of course recommendation systems (CRSs), empirical and practical contributions to expectancy-value theory (EVT), empirical and practical contributions to the field of engineering education, and a practical contribution to the research area of human-centered algorithm design (HCAD).

The main contribution of my thesis is a human-centered methodology for constructing a values-oriented course recommendation system using student informed and EVT-grounded course pathway data. In Chapter 2, I discussed how existing academic course recommendation systems typically rely on academic history, enrollment patterns, and sometimes rely on one loosely related aspect of students' course choice motivations. However, little attention has been given to systematically integrating many motivational constructs into such systems to guide recommendations. My thesis addressed this gap by proposing a methodology for the systematic construction of a motivation-focused and values-oriented dataset of course pathways grounded in HCAD and EVT. My proposed methodology successfully led to the development of an accurate and useful CRS that accounted for 17 aspects of students' values through course choice motivation.

Although EVT has been widely used in engineering education research to examine the motivations behind engineering students' decision making, its constructs have rarely been explicitly translated into learnable features for course recommendation algorithms. The authors of the EVC Light scale suggested that results from their scale could be used to operationalize

such value constructs for machine learning applications regarding predictions of student behaviours in education settings (Schoeffel et al. 2021). My thesis followed this recommendation and contributed an approach for the operationalization of EVT value constructs within the framework of course recommendation that demonstrates the extension of EVT from an explanatory theory to a design resource for course recommendation systems. Through my approach, I also presented one of the first curriculum mappings of EVT's utility and cost.

While EVT has been frequently used to investigate engineering student motivation, few studies examine how motivation constructs manifest across the courses within an engineering curriculum. My thesis provides one of the first exploratory mappings of utility and cost across a computer engineering program, which offers insights into how computer engineering students perceive different courses. My approach to value mapping may lead to novel curriculum analyses and design approaches through comparative examination of how motivational constructs regarding students' course choices evolve across each year in an engineering program, or identification of courses with low utility and high cost that may lead to the redesign of such courses for improved student experiences.

Prior research on course recommendation systems typically design the system at the scale of an institution instead of a specific program. Consequently, research on CRSs designed specifically for engineering students and research on the perspectives of engineering students towards CRSs are both relatively unexplored. My thesis addresses these gaps in the literature through the design and implementation of a CRT specifically for computer engineering students and presentation of the perspectives of computer engineering students towards the CRT. My thesis helps lay the groundwork for future research on engineering specific CRSs, which may lead to increases in engineering students' persistence or improved course planning experiences.

HCAD advocates for stakeholders to participate in the development of algorithms that affect them and for such algorithms to be based in existing social science theories. However, examples of participatory or theoretical dataset construction are limited within CRS research. My thesis demonstrated a participatory approach to dataset construction in which computer engineering students directly contributed to the representations of their motivational values. This representation was grounded in EVT, followed a balance-oriented approach to course selection, and was used alongside theory regarding the benefits of maintained interest. As such, my research is one of the first case studies of applying HCAD to the development of a machine learning dataset.

Evaluations of CRTs typically focus on quantitative performance while few studies have provided insights into students' perceptions of CRTs. This lack of mixed methods evaluations has been discussed as a gap in recommendation systems literature (Bauer et al. 2024). My thesis addressed the gap through presenting a mixed methods evaluation of the usability of my CRT through quantitative performance and the perceptions of computer engineering students. The results from my mixed methods evaluation highlighted the importance of using mixed methods when evaluating CRTs and provide a framework for evaluating values-oriented CRTs.

7.2. Limitations

As this thesis was a pilot project, limitations exist in each phase. In the survey in phase 1 of this project, each course did not receive sufficient responses to robustly represent the utility and cost of each course. The lack of responses was due to the number of participants recruited and the timeframe for gathering data. Despite this, most courses were able to have sufficient robustness

for their representations of utility and cost. Another limitation in phase 1 stems from how interest was represented for each course. Since courses were labelled by whether or not a topic was present in the syllabus or academic calendar description and represented as equal amount of each topic, interest was generalized for each course. A more in-depth representation of interest topics within each course could lead to more representative recommendations for each topic. In phase 2 of my study, I modelled electives in the curriculum into two sub-types for each of the three types of electives, which led to generalization of electives. By either dividing electives into more sub-types or representing each elective as a single course, elective recommendations could become more specific. In phase 3, hardware limitations restricted the hyperparameter exploration space, which was conducted locally. By using a GPU with more than 8 GB of VRAM, or a cloud server, it would be possible to search for larger model configurations, which may have improved performance. In phase 4, only two students participated in the semi-structured interviews due to low participant recruitment within the timeframe of the study. Interviewing more students would improve the robustness of the evaluation and improve understanding of engineering student perceptions of CRSs. Overall, as the scope of this thesis is to examine methodological feasibility of values-oriented course recommendations, these limitations did not have major impacts on the findings. However, these limitations should be considered when using or expanding on the methodologies used in this thesis to improve value representations, recommendation specificity, optimal hyperparameter searching, and evaluation robustness.

7.3. Future Directions

Going forward, I recommend applying my novel human-centered dataset construction methodology to develop CRS for engineering or similarly structured programs, as well as further

investigation into the methodology to for improvements to value representation based in participatory approaches. Similar investigations using different value frameworks can shed light on the robustness of my methodology. The development of similar tools for engineering programs within UM and at other institutions would enable comparative studies on both the underlying machine learning model performance and the impacts on engineering students. As such, a longitudinal study of the experiences and impacts of engineering students using a values-oriented course recommendation tool should be conducted to specifically investigate how the tool effects retention and graduation rates in an engineering program, and engineering students' experiences of longitudinal use of the tool. Additionally, to support future research on CRS I recommend the development of a methodological framework for mixed methods evaluations of CRSs that situate such systems within the contexts of student usability and student experiences. Such a framework would aid in the rigor of CRS evaluations by looking beyond machine learning model performance and evaluating from a holistic and human-centered approach. However, as the ML model used in my CRT was simplistic for the purpose of being a research instrument, increased recommendation performance may be found when investigating advances to the Transformer architecture or other architectural advances for sequence prediction tasks.

7.4. Conclusions

I have several practical recommendations for researchers, engineering departments, academic advisors, and engineering students. For researchers interested in the development of course recommendation systems, I recommend integrating social science research on course choice behaviour and motivation into modelling of students' course choices as such approaches ground the understanding of the tool by grounding inputs and outputs in theory. In this research, I

demonstrated that operationalizing EVT for course recommendations resulted in highly accurate recommendations of values-oriented semesters of courses. As the inputs to my CRT were based on EVT value constructs the generated course pathways were able to be analyzed for their representations of such values. This meant that generated pathways could be better evaluated for their ability to support each value, which leads to better understanding the functionality of the developed CRT. In addition, grounding research on CRSs in social science theories has the potential to guide future comparative research directions. In the case of my thesis, I could change the balance-oriented approach to course selection to a different orientation identified in the literature or use a different values framework instead of EVT. Comparing between these theories can lead to finding which theories work better in what cases of course recommendation.

Based on my CRT being able to successfully make recommendations for a variety of values, I also recommend integrating more than one aspect of course choice motivation into CRSs.

Although it is common to use a single aspect, the complexity of course choice motivation cannot be adequately represented within a single factor. The usability of my CRT highlights that recommending from many factors is feasible and more representative than a single factor. Next, I recommend following participatory approaches of dataset construction. The participatory approach I used to map engineering students' values onto courses in the computer engineering program was crucial for introducing learnable human-centered information into the dataset. My approach enabled students' motivational course choice values to be the direct factor of the information training the underlying machine learning model. Applying my proposed dataset construction methodology may lead to improved course planning experiences of engineering students by increasing students' expectations of successful course planning; reducing the time

and effort of course planning; and maintaining higher levels of important values within students' course pathways.

Lastly, I recommend researchers of CRSs to evaluate such systems through mixed methods approaches that situate the tool within the context of students' use for a more holistic evaluation. This recommendation stems from how much CRS research does not look beyond machine learning model performance and leaves out key perspectives from students who would interact with such a tool. From my evaluation I was able to better understand both the quantitative and qualitative performance of my CRT, both providing insights that the other cannot.

For researchers interested in curriculum analysis, using my survey methodology could lead to findings that improve the balance of utility, cost or the trade-off of utility-cost for preliminary-year students interested in the engineering program at your institution, which could further lead to improved retention. In addition, applying my survey methodology could aid in identifying costly courses and provide opportunity for cost reduction initiatives such as course or curricular redesign based on the perceptions of the engineering students at your institution. Furthermore, application could aid in pathway analysis of institutions where engineering students have less flexibility in their schedules to examine the cost and utility of each semester and open discussion for changes in set pathways for more utility or cost balance.

To engineering departments, I recommend constructing reliable tools and supports for students to plan their course pathways. Course flowcharts and CRTs can aid in the messy and overwhelming experiences of course planning. Findings from my semi-structured interviews highlighted that students value reliable and accurate supports that reduce the time and effort of course planning. Both participants mentioned that their program's course flowchart was their primary support for course planning, which they frequently relied on to be accurate and up to date. One participant

briefly mentioned course flowcharts may not exist for other engineering programs. This is a potential gap in course planning support that could be met by developing a course flowchart or a CRT.

To academic advisors, I recommend using course recommendation tools to aid your understanding of potential course pathways your students could take and encouraging your students to reflect on their expectations and values when planning course pathways. As supports that present pathways to graduation (beyond the course flowchart) are not available, a CRT can fill the gap by presenting course pathways to graduation for students. By encouraging students to reflect, or potentially use a values-oriented CRT if available, academic advisors can help students find course pathways to graduation that align with their values and value trade-offs, which could potentially result in higher retention or graduation rates.

To students, I similarly recommend reflecting on your expectations and values when planning your course pathways to aid in understanding potential value trade-offs and planning your course pathway to align with your values, and to make use of available supports when planning courses for an improved experience. Although the course planning experience is complex, there are a variety of available supports to aid in the process. I also recommend participating in the development of CRSs at your institution such that when you use the CRS, your values are directly factored into the course recommendations you receive. By aligning your use of course planning supports with your values, you may experience increased motivation or improved expectations of academic achievement and successful course planning. It should further be noted that when using a CRS, the CRS is limited by its curricular modelling, meaning your academic advisor can aid you in overriding program constraints to better align your course pathway with your academic goals.

7.5. Learning Reflection

Finally, in this section I reflect on my learning experience throughout my research project. There are key aspects that shaped my learning experience and my thesis – all of which center on the benefits of applying social science to machine learning. Through my research, I learned that social science can be operationalized as useful input features for ML models, despite being often overlooked. From my positionality, I learned that my previous understanding of ML was entirely derived from quantitative approaches. My research led me to learn about methods for integrating social science into ML and the benefits of doing so from both social science and ML perspectives.

A major learning for me was that developing ML models through theoretical frameworks aids in understanding model outputs. Before my research project I had no understanding of what a theoretical framework was or how to use one. ML research tends to ignore theoretical frameworks and focus solely on performance improvements through positive lenses. Although theoretical frameworks are quite intuitive to use, what I did not expect to learn was that using theoretical frameworks improves the interpretability of the output of the ML model. Literature on the framework can guide interpretation of what the model output means with respect to its corresponding inputs. This interpretability can aid in understanding and debugging the model; and also aids in more contextually communicating the performance and outputs of the model.

I also learned that designing a tool grounded in literature helps better understand what aspects of such literature are being researched and what aspects are being used as research instruments. Initially, I framed my research as a case study of developing a CRT for engineering students. However, when presenting the tool to others, I noticed I was discussing the underlying machine learning model and how I set up the model for my research. Although I put in much effort to set

up the model, it was not until I grounded my research in literature that I realized that the machine learning model was simply the research instrument I needed to assess the feasibility of my approach to creating a values-oriented dataset of course pathways for course recommendation. Much ML research does not thoroughly ground their designs in the contextual literature around their problem formulation. This often leads to designing purely for quantitative model performance without literature supporting design choices and stakeholder impacts. From my research I found that through grounding my design in literature, I better understood what other aspects from the literature could be changed for similar designs that take a different theoretical approach i.e. design modularity. This clearly separated the course choice orientation, value framework, and machine learning model in my research as modular components that tell a story of what I am considering and how I am assessing my CRT.

In the final phase of my research, I learned that mixed methods evaluations of ML-based tools are crucial to a holistic evaluation. Qualitative evaluations of such tools are often overlooked yet provide meaningful findings for understanding the tool and future directions for research and development. As I previously mentioned, before I conducted this research project, I was primarily concerned with quantitative evaluations of machine learning models as it is simple to compare numerical performance between models to evaluate which model has the best performance. Although this still holds, it was clear in the literature that the use of qualitative evaluations alongside quantitative evaluations were seen as the more holistic approach to evaluation since quantitative results, in isolation, often struggle to capture the complex impacts ML models may have on stakeholders. In my research, I was able to see that the perspectives of participants in my user study provided meaningful insights on the evaluation of my CRT and suggestions of improvements to my CRT that quantitative data could never provide. Overall, my

learning experience from this research broadened and added considerable depth to my understanding and appreciation of combining social science with ML for human-centered AI applications.

Ethics Statement

All procedures involving human participants conducted as part of this thesis were approved by and implemented in accordance with the University of Manitoba Research Ethics Board. The protocol number for the survey is HE2025-0246 and the protocol for the semi-structured interviews is HE2026-0021. All participants remained anonymous in these studies. All associated data will remain confidential and securely stored in keeping with the University of Manitoba Research Ethics Board requirements. See Appendix B for the consent form of each study.

References

- Adomavicius, G. and J. Zhang. 2012. Impact of data characteristics on recommender systems performance. *ACM Transactions on Management Information Systems* 3(1): 1–17. <https://doi.org/10.1145/2151163.2151166>.
- Akiba, T., S. Sano, T. Yanase, T. Ohta and M. Koyama. 2019. Optuna: A Next-generation Hyperparameter Optimization Framework. arXiv, 25 July 2019. <https://doi.org/10.48550/arXiv.1907.10902> (2025 /12/22)
- Algarni, S. and F. Sheldon. 2023. Systematic Review of Recommendation Systems for Course Selection. *Machine Learning and Knowledge Extraction* 5(2): 560–596. <https://doi.org/10.3390/make5020033>.
- Alkin, M. 2004. *Evaluation Roots*. 2455 Teller Road, Thousand Oaks California 91320 United States of America: SAGE Publications, Inc. <https://doi.org/10.4135/9781412984157>
- Arlot, S. and A. Celisse. 2010. A survey of cross-validation procedures for model selection. *Statistics Surveys* 4(none). <https://doi.org/10.1214/09-SS054>
- Atalla, S., M. Daradkeh, A. Gawanmeh, H. Khalil, W. Mansoor, S. Miniaoui and Y. Himeur. 2023. An Intelligent Recommendation System for Automating Academic Advising Based on Curriculum Analysis and Performance Modeling. *Mathematics* 11(5): 1098. <https://doi.org/10.3390/math11051098>.
- Babad, E. and A. Tayeb. 2003. Experimental analysis of students' course selection. *British Journal of Educational Psychology* 73: 373–393.
- Barron, K., C. Hulleman, S. Getty and J. Taylor. 2017. Expectancy-value-cost survey of student motivation: User guide. <https://www.researchgate.net/publication/319433479>.
- Bauer, C., A. Said and E. Zangerle. 2024. *Evaluation Perspectives of Recommender Systems: Driving Research and Education (Dagstuhl Seminar 24211)*. Dagstuhl Reports (DagRep)14 5. Schloss Dagstuhl – Leibniz-Zentrum für Informatik. <https://doi.org/10.4230/DAGREP.14.5.58> (2026 /2/6)
- Baumer, E.P. 2017. Toward human-centered algorithm design. *Big Data & Society* 4(2): 205395171771885. <https://doi.org/10.1177/2053951717718854>.
- Bego, C., P. Thomas, X. Wang and A. Dourado. 2022. Investigating Engineering Persistence through Expectancy Value Theory and Machine Learning Techniques. In *2022 ASEE Annual Conference & Exposition Proceedings*, 41559. Minneapolis, MN: ASEE Conferences. <https://doi.org/10.18260/1-2--41559> (2026 /1/12)
- Bergstra, J. and Y. Bengio. 2012. Random Search for Hyper-Parameter Optimization. *Journal of Machine Learning Research* 13: 281–305.
- Borchers, C. and Z.A. Pardos. 2023. Insights into undergraduate pathways using course load analytics. In *LAK23: 13th International Learning Analytics and Knowledge Conference*, 219–229. Arlington TX USA: ACM. <https://doi.org/10.1145/3576050.3576081> (2025 /4/19)
- Borgonovo, E. 2007. A new uncertainty importance measure. *Reliability Engineering & System Safety* 92(6): 771–784. <https://doi.org/10.1016/j.ress.2006.04.015>.
- Borgonovo, E. and E. Plischke. 2016. Sensitivity analysis: A review of recent advances. *European Journal of Operational Research* 248(3): 869–887. <https://doi.org/10.1016/j.ejor.2015.06.032>.
- Brody, S., U. Alon and E. Yahav. 2023. On the Expressivity Role of LayerNorm in Transformers' Attention. In *Findings of the Association for Computational Linguistics: ACL 2023*,

- 14211–14221. Toronto, Canada: Association for Computational Linguistics.
<https://doi.org/10.18653/v1/2023.findings-acl.895> (2025 /12/15)
- Chen, Y., A. Fu, J.J.-L. Lee, I.W. Tomasik and R.F. Kizilcec. 2022. Pathways: Exploring Academic Interests with Historical Course Enrollment Records. In *Proceedings of the Ninth ACM Conference on Learning @ Scale*, 222–233. New York City NY USA: ACM.
<https://doi.org/10.1145/3491140.3528270> (2025 /4/18)
- Chollet, F. 2015. Keras. <https://github.com/fchollet/keras>.
- Ciampiconi, L., A. Elwood, M. Leonardi, A. Mohamed and A. Rozza. 2024. A survey and taxonomy of loss functions in machine learning arXiv, 18 November 2024.
<https://doi.org/10.48550/arXiv.2301.05579> (2025 /12/28)
- Cronbach, L. 1951. Coefficient alpha and the internal structure of tests. *PSYCHOMETRIKA* 16(3).
- Cui, Y., F. Chen, A. Shiri and Y. Fan. 2019. Predictive analytic models of student success in higher education: A review of methodology. *Information and Learning Sciences* 120(3/4): 208–227. <https://doi.org/10.1108/ILS-10-2018-0104>.
- Dean, J., R. Monga and Google Research. 2015. TensorFlow: Large-Scale Machine Learning on Heterogeneous Distributed Systems. <https://www.tensorflow.org/>.
- DeepSeek-AI. 2024. DeepSeek LLM: Scaling Open-Source Language Models with Longtermism. arXiv, 5 January 2024. <https://doi.org/10.48550/arXiv.2401.02954> (2026 /8/8)
- Engineers Canada. 2025a. Computer engineering syllabus. <https://engineerscanada.ca/regulatory-excellence/examination-syllabi/computer-engineering-syllabus> (2025 /8/17)
- Engineers Canada. 2025b. Basic studies syllabus. <https://engineerscanada.ca/regulatory-excellence/examination-syllabi/basic-studies-syllabus> (2025 /8/17)
- Esteban, A., A. Zafra and C. Romero. 2018. A Hybrid Multi-Criteria approach using a Genetic Algorithm for Recommending Courses to University Students. *Proceedings of the 11th International Conference on Educational Data Mining*.
- Fatahi, B., H. Khabbaz, J. Xue and R. Hadgraft. 2023. Generative AI as a Catalyst for Enhanced Learning Experience in Engineering Education. In *Proceedings of 34th Australasian Association for Engineering Education Conference (AAEE)*.
- Fawcett, T. 2006. An introduction to ROC analysis. *Pattern Recognition Letters* 27(8): 861–874.
<https://doi.org/10.1016/j.patrec.2005.10.010>.
- Fosnacht, K., A.C. McCormick, J.N. Nailos and A.K. Ribera. 2017. Frequency of First-Year Student Interactions With Advisors. *NACADA Journal* 37(1): 74–86.
<https://doi.org/10.12930/NACADA-15-048>.
- Friedman, B. 1996. Value-sensitive design. *Interactions* 3(6): 16–23.
<https://doi.org/https://doi.org/10.1145/242485.242493>.
- Friedman, B. and D.G. Hendry. 2019. *Value Sensitive Design: Shaping Technology with Moral Imagination*. The MIT Press. <https://doi.org/10.7551/mitpress/7585.001.0001>
- Fushiki, T. 2011. Estimation of prediction error by using K-fold cross-validation. *Statistics and Computing* 21(2): 137–146. <https://doi.org/10.1007/s11222-009-9153-8>.
- Gerdes, A. and T.F. Frandsen. 2023. A systematic review of almost three decades of value sensitive design (VSD): what happened to the technical investigations? *Ethics and Information Technology* 25(2): 26. <https://doi.org/10.1007/s10676-023-09700-2>.

- Geva, M., R. Schuster, J. Berant and O. Levy. 2021. Transformer Feed-Forward Layers Are Key-Value Memories. arXiv, 5 September 2021. <https://doi.org/10.48550/arXiv.2012.14913> (2025 /4/23)
- Guillén-Yparrea, N. and F. Hernández-Rodríguez. 2024. Unveiling Generative AI in Higher Education: Insights from Engineering Students and Professors. In *2024 IEEE Global Engineering Education Conference (EDUCON)*, 1–5. Kos Island, Greece: IEEE. <https://doi.org/10.1109/EDUCON60312.2024.10578876> (2024 /10/29)
- Guruge, D.B., R. Kadel and S.J. Halder. 2021. The State of the Art in Methodologies of Course Recommender Systems—A Review of Recent Research. *Data* 6(2): 18. <https://doi.org/10.3390/data6020018>.
- Harris, H. and J. Kittur. 2024. Generative Artificial Intelligence in Undergraduate Engineering: A Systematic Literature Review. In *2024 ASEE Annual Conference & Exposition Proceedings*, 47492. Portland, Oregon: ASEE Conferences. <https://doi.org/10.18260/1-2--47492> (2024 /10/29)
- Hartson, H.R. 1998. Human–computer interaction: Interdisciplinary roots and trends. *Journal of Systems and Software* 43(2): 103–118. [https://doi.org/10.1016/S0164-1212\(98\)10026-2](https://doi.org/10.1016/S0164-1212(98)10026-2).
- Herman, J. and W. Usher. 2017. SALib: An open-source Python library for Sensitivity Analysis. *The Journal of Open Source Software* 2(9): 97. <https://doi.org/10.21105/joss.00097>.
- Hidi, S. and K.A. Renninger. 2006. The Four-Phase Model of Interest Development. *Educational Psychologist* 41(2): 111–127. https://doi.org/10.1207/s15326985ep4102_4.
- Hidi, S.E. and K.A. Renninger. 2026. The four-phase model of interest development: Addressing individual differences. *Learning and Individual Differences* 126: 102865. <https://doi.org/10.1016/j.lindif.2025.102865>.
- Honken, N., P. Ralston and T. Tretter. 2016. Step-Outs to Stars: Engineering Retention Framework. In *2016 ASEE Annual Conference & Exposition Proceedings*, 25877. New Orleans, Louisiana: ASEE Conferences. <https://doi.org/10.18260/p.25877> (2026 /5/14)
- Hsieh, H.-F. and S.E. Shannon. 2005. Three Approaches to Qualitative Content Analysis. *Qualitative Health Research* 15(9): 1277–1288. <https://doi.org/10.1177/1049732305276687>.
- Jiawei Li, Qianru Lyu, Wei Qiu and Andy W. H. Khong. 2025. Improving Course Recommendation Systems with Explainable AI: LLM-Based Frameworks and Evaluations. <https://doi.org/10.5281/ZENODO.15870185>
- Ju, S.-J. 2009. On the distribution type of uncertain inputs for probabilistic assessment. *Reliability Engineering & System Safety* 94(5): 964–968. <https://doi.org/10.1016/j.ress.2008.11.001>.
- Khali, A. and J. Williamson. 2014. Role of Academic Advisors in the Success of Engineering Students. *Universal Journal of Educational Research* 2(1): 73–79. <https://doi.org/10.13189/ujer.2014.020109>.
- Kirkham, J., E. Chapman and S. Male. 2023. Measuring Motivation for Mathematics Course Choice in Secondary School Students: Interrelationships Between Cost and Other Situated Expectancy-Value Theory Components. *Sage Open* 13(2): 21582440231180671. <https://doi.org/10.1177/21582440231180671>.
- Kosovich, J.J., C.S. Hulleman, K.E. Barron and S. Getty. 2015. A Practical Measure of Student Motivation: Establishing Validity Evidence for the Expectancy-Value-Cost Scale in Middle School. *The Journal of Early Adolescence* 35(5–6): 790–816. <https://doi.org/10.1177/0272431614556890>.

- Lai, L.-H., Y.-L. Lin, Y.-H. Liu, J.-P. Lai, W.-C. Yang, H.-P. Hou and P.-F. Pai. 2024. The Use of Machine Learning Models with Optuna in Disease Prediction. *Electronics* 13(23): 4775. <https://doi.org/10.3390/electronics13234775>.
- Lam, C. and D. Wilson. 2024. Value and Interest: Do They Really Make a Difference in Student Engagement. In *2024 ASEE Annual Conference & Exposition Proceedings*, 48253. Portland, Oregon: ASEE Conferences. <https://doi.org/10.18260/1-2--48253> (2026 /1/12)
- Lee, H.R., L. Von Keyserlingk, R. Arum and J.S. Eccles. 2021. Why Do They Enroll in This Course? Undergraduates' Course Choice From a Motivational Perspective. *Frontiers in Education* 6: 641254. <https://doi.org/10.3389/educ.2021.641254>.
- Lessmann, K., A. Beyer, J. Klasmeier and M. Matthies. 2005. Influence of Distributional Shape of Substance Parameters on Exposure Model Output. *Risk Analysis* 25(5): 1137–1145. <https://doi.org/10.1111/j.1539-6924.2005.00669.x>.
- Ma, B., M. Lu, Y. Taniguchi and S. Konomi. 2021. Investigating course choice motivations in university environments. *Smart Learning Environments* 8(1): 31. <https://doi.org/10.1186/s40561-021-00177-4>.
- Matusovich, H.M., R.A. Streveler and R.L. Miller. 2010. Why Do Students Choose Engineering? A Qualitative, Longitudinal Investigation of Students' Motivational Values. *Journal of Engineering Education* 99(4): 289–303. <https://doi.org/10.1002/j.2168-9830.2010.tb01064.x>.
- McConvey, K., S. Guha and A. Kuzminykh. 2023. A Human-Centered Review of Algorithms in Decision-Making in Higher Education. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems*, 1–15. <https://doi.org/10.1145/3544548.3580658> (2025 /4/18)
- Mertens, D.M. 2018. *Mixed Methods Design in Evaluation*. 2455 Teller Road, Thousand Oaks California 91320: SAGE Publications, Inc. <https://doi.org/10.4135/9781506330631>
- Mikolov, T., K. Chen, G. Corrado and J. Dean. 2013. Efficient Estimation of Word Representations in Vector Space. arXiv, 7 September 2013. <https://doi.org/10.48550/arXiv.1301.3781> (2025 /4/20)
- Molontay, R., N. Horvath, J. Bergmann, D. Szekrenyes and M. Szabo. 2020. Characterizing Curriculum Prerequisite Networks by a Student Flow Approach. *IEEE Transactions on Learning Technologies* 13(3): 491–501. <https://doi.org/10.1109/TLT.2020.2981331>.
- Moon, K.Z., U. Salma and M.S. Uddin. 2024. BERT-Based Personalized Course Recommendation System from Online Learning Platform. In *2024 6th International Conference on Electrical Engineering and Information & Communication Technology (ICEEICT)*, 980–985. Dhaka, Bangladesh: IEEE. <https://doi.org/10.1109/ICEEICT62016.2024.10534490> (2026 /1/9)
- Morsy, S. and G. Karypis. 2018. Learning Course Sequencing for Course Recommendation. *University Digital Conservancy*. <https://hdl.handle.net/11299/216025>.
- Nam, K. and J. Kim. 2025. Determination and application of hyperparameters for landslide susceptibility assessment using Optuna: the Eunsan shallow landslides occurred on August 14, 2022, in Buyeo City, Chungcheong Province, Korea. *Geosciences Journal* 29(6): 978–994. <https://doi.org/10.1007/s12303-025-00054-z>.
- OpenAI. 2024. GPT-4 Technical Report. arXiv, 4 March 2024. <https://doi.org/10.48550/arXiv.2303.08774> (2025 /4/20)
- Ortega-Alvarez, J., S.Z. Atiq and H. Rodriguez-Simmonds. 2016. A Qualitative Study Investigating How First-Year Engineering Students' Value Beliefs Influence their Choice

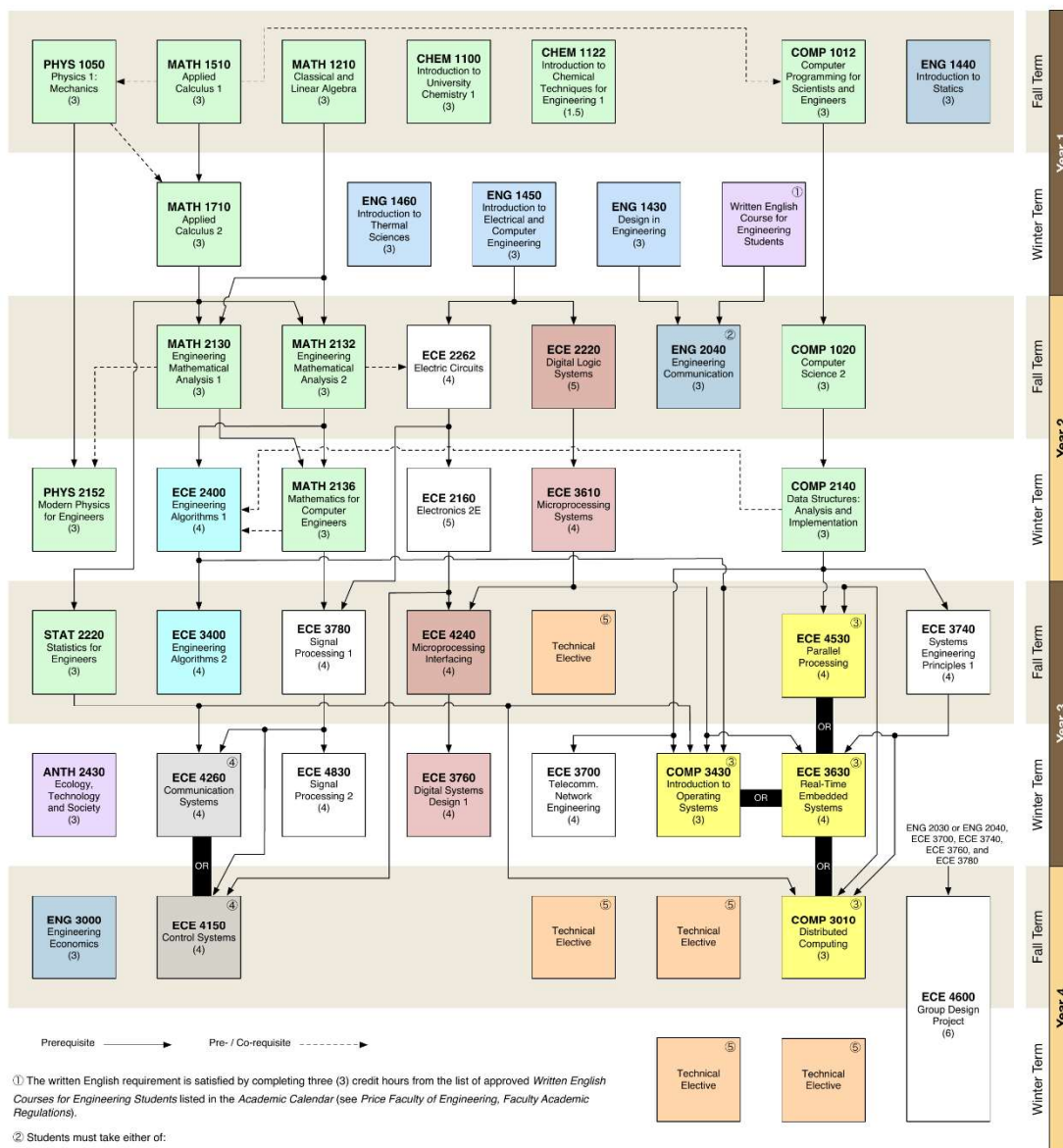
- of Selecting an Engineering Major. In *2016 ASEE Annual Conference & Exposition Proceedings*, 26417. New Orleans, Louisiana: ASEE Conferences. <https://doi.org/10.18260/p.26417> (2025 /4/24)
- Pardos, Z.A., Z. Fan and W. Jiang. 2018. Connectionist Recommendation in the Wild: On the utility and scrutability of neural networks for personalized course guidance. *arXiv*, 31 December 2018. <https://doi.org/10.48550/arXiv.1803.09535> (2025 /4/19)
- Pardos, Z.A. and W. Jiang. 2020. Designing for serendipity in a university course recommendation system. In *Proceedings of the Tenth International Conference on Learning Analytics & Knowledge*, 350–359. Frankfurt Germany: ACM. <https://doi.org/10.1145/3375462.3375524> (2025 /4/19)
- Petrik, O., J.D.A.E. Silva and F.M.M.V.E. Moura. 2013. Impact of Distribution Choice for Representing Input Variation: Analysis of Uncertainty in Travel Demand Simulation in Context of Information Shortage. *Transportation Research Record: Journal of the Transportation Research Board* 2344(1): 40–48. <https://doi.org/10.3141/2344-05>.
- Pintrich, P.R. 2003. A Motivational Science Perspective on the Role of Student Motivation in Learning and Teaching Contexts. *Journal of Educational Psychology* 95(4): 667–686. <https://doi.org/10.1037/0022-0663.95.4.667>.
- Plischke, E., E. Borgonovo and C.L. Smith. 2013. Global sensitivity measures from given data. *European Journal of Operational Research* 226(3): 536–550. <https://doi.org/10.1016/j.ejor.2012.11.047>.
- Prokhorov, V., E. Shareghi, Y. Li, M.T. Pilehvar and N. Collier. 2019. On the Importance of the Kullback-Leibler Divergence Term in Variational Autoencoders for Text Generation. *arXiv*, 30 September 2019. <https://doi.org/10.48550/arXiv.1909.13668> (2025 /12/17)
- Qadir, J. 2023. Engineering Education in the Era of ChatGPT: Promise and Pitfalls of Generative AI for Education. In *2023 IEEE Global Engineering Education Conference (EDUCON)*, 1–9. Kuwait, Kuwait: IEEE. <https://doi.org/10.1109/EDUCON54358.2023.10125121> (2024 /10/29)
- Radford, A., K. Narasimhan, T. Salimans and I. Sutskever. 2018. Improving Language Understanding by Generative Pre-Training. *OpenAI*.
- Ricci, F., L. Rokach, B. Shapira and P.B. Kantor (Eds). 2011. *Recommender Systems Handbook*. Boston, MA: Springer US. <https://doi.org/10.1007/978-0-387-85820-3>
- Richardson, E., R. Trevizani, J.A. Greenbaum, H. Carter, M. Nielsen and B. Peters. 2024. The receiver operating characteristic curve accurately assesses imbalanced datasets. *Patterns* 5(6): 100994. <https://doi.org/10.1016/j.patter.2024.100994>.
- Robinson, B., T. Tretter, J. Lewis and N. Hawkins. 2023. Capturing First-Year Engineering Students' Situational and Individual Interest via a Formal Makerspace Course. In *2023 ASEE Annual Conference & Exposition Proceedings*, 43120. Baltimore, Maryland: ASEE Conferences. <https://doi.org/10.18260/1-2--43120> (2025 /5/17)
- Ryan, R.M. and E.L. Deci. 2000. Self-Determination Theory and the Facilitation of Intrinsic Motivation, Social Development, and Well-Being. *American Psychologist*. <https://doi.org/10.1037/110003-066X.55.1.68>.
- Saltelli, A. 2002. Making best use of model evaluations to compute sensitivity indices. *Computer Physics Communications* 145(2): 280–297. [https://doi.org/10.1016/S0010-4655\(02\)00280-1](https://doi.org/10.1016/S0010-4655(02)00280-1).

- Saltelli, A., P. Annoni, I. Azzini, F. Campolongo, M. Ratto and S. Tarantola. 2010. Variance based sensitivity analysis of model output. Design and estimator for the total sensitivity index. *Computer Physics Communications* 181(2): 259–270. <https://doi.org/10.1016/j.cpc.2009.09.018>.
- Schoeffel, P., V.F.C. Ramos, C. Cechinel and R.S. Wazlawick. 2021. The Expectancy-Value-Cost Light Scale to Measure Motivation of Students in Computing Courses. *Informatics in Education*. <https://doi.org/10.15388/infedu.2022.04>
- Shadish, W.R. 1998. Evaluation Theory is Who We Are. *American Journal of Evaluation* 19(1): 1–19. <https://doi.org/10.1177/109821409801900102>.
- Shao, E., S. Guo and Z.A. Pardos. 2021. Degree Planning with PLAN-BERT: Multi-Semester Recommendation Using Future Courses of Interest. *Proceedings of the AAAI Conference on Artificial Intelligence* 35(17): 14920–14929. <https://doi.org/10.1609/aaai.v35i17.17751>.
- Shekhar, S., A. Bansode and A. Salim. 2022. A Comparative study of Hyper-Parameter Optimization Tools. arXiv, 17 January 2022. <https://doi.org/10.48550/arXiv.2201.06433> (2025 /12/28)
- Silveira, T., M. Zhang, X. Lin, Y. Liu and S. Ma. 2019. How good your recommender system is? A survey on evaluations in recommendation. *International Journal of Machine Learning and Cybernetics* 10(5): 813–831. <https://doi.org/10.1007/s13042-017-0762-9>.
- Singh, J., L.A. Kumaraswamidhas, N. Bura and N. Dilawar Sharma. 2021. A Monte Carlo simulation investigation on the effect of the probability distribution of input quantities on the effective area of a pressure balance and its uncertainty. *Measurement* 172: 108853. <https://doi.org/10.1016/j.measurement.2020.108853>.
- Singh, P., A. Necholi and W. Moreno. 2023. Synthetic Data Generation for Engineering Education: A Bayesian Approach. In *2023 IEEE 3rd International Conference on Advanced Learning Technologies on Education & Research (ICALTER)*, 1–4. <https://doi.org/10.1109/ICALTER61411.2023.10372935> (2025 /1/31)
- Sipper, M. 2022. High Per Parameter: A Large-Scale Study of Hyperparameter Tuning for Machine Learning Algorithms. *Algorithms* 15(9): 315. <https://doi.org/10.3390/a15090315>.
- Sobol, I.M. 2001. Global sensitivity indices for nonlinear mathematical models and their Monte Carlo estimates. *Mathematics and Computers in Simulation* 55(1–3): 271–280. [https://doi.org/10.1016/S0378-4754\(00\)00270-6](https://doi.org/10.1016/S0378-4754(00)00270-6).
- Spiekermann, S. and T. Winkler. 2020. Value-based Engineering for Ethics by Design. *SSRN Electronic Journal*. <https://doi.org/10.2139/ssrn.3598911>
- Taber, K.S. 2018. The Use of Cronbach's Alpha When Developing and Reporting Research Instruments in Science Education. *Research in Science Education* 48(6): 1273–1296. <https://doi.org/10.1007/s11165-016-9602-2>.
- Tavakol, M. and R. Dennick. 2011. Making sense of Cronbach's alpha. *International Journal of Medical Education* 2: 53–55. <https://doi.org/10.5116/ijme.4dfb.8dfd>.
- Team Gemini. 2025. Gemini: A Family of Highly Capable Multimodal Models. arXiv, 9 May 2025. <https://doi.org/10.48550/arXiv.2312.11805> (2026 /8/8)
- Thomas, P., C. Bego and A. Dourado. 2023. Predicting Student Retention via Expectancy Value Theory Using Data Gathered before the Semester Begins. In *2023 ASEE Annual Conference & Exposition Proceedings*, 43930. Baltimore, Maryland: ASEE Conferences. <https://doi.org/10.18260/1-2--43930> (2026 /1/12)

- Vaswani, A., N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A.N. Gomez, L. Kaiser and I. Polosukhin. 2017. Attention Is All You Need. arXiv, 2017.
<http://arxiv.org/abs/1706.03762> (2024 /10/29)
- Wigfield, A. and J.S. Eccles. 2000. Expectancy–Value Theory of Achievement Motivation. *Contemporary Educational Psychology* 25(1): 68–81.
<https://doi.org/10.1006/ceps.1999.1015>.
- Wu, F., W. Fan, C. Arbona and D. De La Rosa-Pohl. 2020. Self-efficacy and subjective task values in relation to choice, effort, persistence, and continuation in engineering: an Expectancy-value theory perspective. *European Journal of Engineering Education* 45(1): 151–163. <https://doi.org/10.1080/03043797.2019.1659231>.
- Xiong, R., Y. Yang, D. He, K. Zheng, S. Zheng, C. Xing, H. Zhang, Y. Lan, L. Wang and T.-Y. Liu. 2020. On Layer Normalization in the Transformer Architecture arXiv, 29 June 2020.
<https://doi.org/10.48550/arXiv.2002.04745> (2025 /12/15)
- Xu, J., X. Sun, Z. Zhang, G. Zhao and J. Lin. 2019. Understanding and Improving Layer Normalization. *33rd Conference on Neural Information Processing Systems*.
<https://doi.org/https://doi.org/10.48550/arXiv.1911.07013>.
- Xu, J., C. Wu, Y.-F. Li, G. Danoy and P. Bouvry. 2025a. A Unified Hyperparameter Optimization Pipeline for Transformer-Based Time Series Forecasting Models.
<https://doi.org/https://doi.org/10.48550/arXiv.2501.01394>.
- Xu, J.Z., Y. Patel, S. Mohammad and C. Variawa. 2025b. Using Expectancy Value Theory to Explore the Impact of Assessment Characteristics on Student Ownership of Learning. *Proceedings of the Canadian Engineering Education Association (CEEA)*.
<https://doi.org/https://doi.org/10.24908/pceea.2025.19601>.
- Yang, L. and A. Shami. 2020. On Hyperparameter Optimization of Machine Learning Algorithms: Theory and Practice. *Neurocomputing* 415: 295–316.
<https://doi.org/10.1016/j.neucom.2020.07.061>.
- Zhang, T. 2009. Human-Centred Design: An Emergent Conceptual Model. *Royal College of Art*.
<https://bura.brunel.ac.uk/handle/2438/3472>.
- Zhang, Y. and B.M. Wildemuth. 2009. Qualitative Analysis of Content. *Applications of Social Research Methods to Questions in Information and Library Science*.

Appendix A: UM Computer Engineering Flowchart 2025 – 2026

2025 – 2026 Computer Engineering Course Flow Chart
Model 4 Year Program



① The written English requirement is satisfied by completing three (3) credit hours from the list of approved *Written English Courses for Engineering Students* listed in the *Academic Calendar* (see *Price Faculty of Engineering, Faculty Academic Regulations*).

② Students must take either of:
– ENG 2030 *Engineering Communication: Strategies for the Profession*
– ENG 2040 *Engineering Communication: Strategies, Practice, and Design*

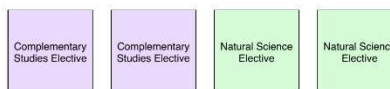
③ Students must take one of:
– ECE 3630 *Real-time Embedded Systems* – COMP 3010 *Distributed Computing*
– ECE 4530 *Parallel Processing* – COMP 3430 *Operating Systems*

④ Students must take either of:
– ECE 4150 *Control Systems*
– ECE 4260 *Communication Systems*

⑤ Technical Electives:
– Five (5) technical electives are required to complete the program.
– At most two (2) of these electives may be selected from the list of approved Electrical Engineering courses.
– Technical electives may be taken at anytime, subject to prerequisites.

This flow chart is intended as a guide, and only applies for the current academic year. It should not be used as a guide for subsequent years. Errors may be present in this document. Students should refer to information in the *Academic Calendar*.

Additional required elective courses which may be completed in any term.



Appendix B: Consent Forms

Consent Form: Survey of Computer Engineering Students' Value Perceptions of Computer Engineering Courses

Participant Informed Consent Form

Study Title: Survey of Computer Engineering Students' Value Perceptions of Computer Engineering Courses.

Student Principal Investigator: Luke Slade, Master's Student, Biosystems Engineering, Faculty of Agricultural and Food Sciences, sladel@myumanitoba.ca.

Student Advisor: Kari Zacharias, Assistant Professor, Engineering Education, Price Faculty of Engineering, kari.zacharias@umanitoba.ca.

Conflicts of Interest and Undue Influence

Luke Slade and Kari Zacharias do not have any conflicts of interest in this study.

Invitation to Participate

You are being invited to participate in a research study. This consent form, a copy of which you can download or print, is only part of the process of informed consent. If you want more details about something mentioned here, or information not included here, feel free to ask Luke Slade whose contact information can be found above. Please take the time to read this document and any accompanying information carefully. It is very important that you understand:

- what is being asked of you,
- what the risks and benefits of participation are, and
- how the information you provide will be used and stored.

Purpose of the Study

I am doing this research for a thesis under the supervision of Kari Zacharias. This research study is about mapping the utility value (how useful something is) and cost value (what is given up doing something) onto courses in the computer engineering program at University of Manitoba. I am hoping to create a way to evaluate course pathway recommendation tools (tools that recommend what courses to take next) in how well they align recommendations with the values of computer engineering students.

Study Procedures

If you decide to take part in the study, you will be asked to complete an online survey through Google Forms. The survey should take about 20 minutes to complete.

Study Risks

This survey is minimal risk, meaning that the risks are no greater than what you might encounter in your daily life.

Study Benefits

There are no direct benefits to you by participating in this study, but we hope to learn more about how courses in the computer engineering program are associated with student values so we can develop better course recommendation tools that account for student values in the future.

Compensation

Upon completion of the survey, you can voluntarily enter for a chance to win a \$50 amazon gift card. There will be one gift card winner drawn from all survey participants. Participants are not required to answer all questions in order to participate in the draw for the gift card, i.e. they can skip questions. Since this study aims to recruit between 12 and 20 participants, there is an 8% - 5% chance of winning. More or less participants may be recruited which will affect these odds of winning. This draw will be conducted by a third party. Participation in this draw is voluntary, and emails required for the draw will be collected separately from the survey responses.

Confidentiality

All the information you provide as a participant in this study is confidential which means we as researchers must keep it safe. We will do our best, however, it is not possible to guarantee absolute confidentiality. We will only share your personal information if required through court order or law.

Use and Storage of Information

Your information will initially be stored in a private Google Drive folder, accessible only to the SPI and Advisor, and will be transferred for storage to a private and secure Microsoft Teams folder approved by the University of Manitoba, accessible only to the SPI and advisor, for long-term storage, after which all information in the Google Drive will be deleted. The research team will see your responses.

When we share the results of this study, we will combine everybody's responses. This means no one will see your answers.

No identifiable information will be collected as part of the survey. Contact information will be collected separately from the survey to participate in the gift card draw, to potentially participate in future phases of the study, and to receive a summary of results. We will destroy your contact information for the draw, for participation in future phases, and for receiving a summary of results by 04/2026. Your responses may be kept indefinitely.

Data Sharing

Your research information may be shared outside of the University of Manitoba with researchers, other organizations, and/or made publicly available. The information is being shared for further analysis or testing, as part of the research study, and/or because it is required by a funder or journal. It will not include your name or any information that could directly identify you.

Dissemination

The study results will be shared through

1. the publication of a master's thesis;
2. journal publications and presentations in relevant academic fields; and/or
3. a summarized report of the study results to be sent to interested participants.

Withdrawing

Your participation in this research study is voluntary. You can choose to answer only the questions that you are comfortable with. You may withdraw from the study for any reason. However, if you choose to withdraw, you must do so before you submit your responses. You can do this by exiting the browser. You do not have to explain why. You will not be penalized in any way. If you withdraw partway through the study, all of your responses will be destroyed. Once you submit your responses, you cannot be withdrawn from the study because we will not be able to link your responses back to you.

Summary of Results

A summary of the study results will be available by 04/2026. If you wish to receive a copy by email, please provide your email address at the end of the survey.

Questions or Concerns

Designated University of Manitoba personnel may check that this study is being done safely and properly. To do this, they may review the research records.

This study has been reviewed and approved by a Research Ethics Board at the University of Manitoba. However, this does not mean that participation is risk-free. If you have questions about your rights as a research participant, you may contact the Office of Human Research Ethics at humanethics@umanitoba.ca or (204) 474-7122.

If you have any questions, concerns, or complaints about this study, you may contact any members of the research team listed on the first page or the Office of Human Research Ethics.

Consent:

By participating in this study, I agree that:

- I have read the above information.
- I have had the opportunity to ask and have answered all of my questions.
- I understand what is being asked of me.
- I will be taking part in a research study.
- I may freely stop or leave the research study activities at any time.
- My information may be shared outside the University of Manitoba.
- I do not waive my legal rights by participating in the study.

Notice Regarding Collection, Use, and Disclosure of Personal Information

Your personal information is being collected under the authority of *The University of Manitoba Act*. The University of Manitoba is committed to preserving your right to privacy. The information you provide will be

used by the University to support our research. Your personal information will not be used or disclosed for other purposes, unless permitted by *The Freedom of Information and Protection of Privacy Act* or *The Personal Health Information Act*. If you have any questions about the collection of personal information:

Ph: 204-474-9462 or Email: fippa@umanitoba.ca

The University of Manitoba wants you to know that the information you share with our researchers will only be used in the ways explained in this form and as Manitoba privacy laws allow. If you have questions, you can call 204-474-9462 or Email: fippa@umanitoba.ca.

I agree to participate in this study.	☐ YES ☐ NO
I agree to be contacted by a member of this research team for future phases related to this research study. If yes, please provide an email address/contact information: _____ <i>By agreeing to be contacted, you are not required to participate in any future phases. We will keep your contact information for the duration of this multi-phase study. If you wish to withdraw your name from this list, please contact Luke Slade, sladel@myumanitoba.ca.</i>	☐ YES ☐ NO

I would like to receive a summary of the results of this study. If yes, please provide an email address/contact information: _____	☐ YES ☐ NO
---	---

Consent Form: Interviews of Computer Engineering Students for their Perceptions of a Course Recommendation Tool

Participant Informed Consent Form

Study Title: Interviews of Computer Engineering Students for their Perceptions of a Course Recommendation Tool.

Student Principal Investigator: Luke Slade, Master's Student, Biosystems Engineering, Faculty of Agricultural and Food Sciences, sladel@myumanitoba.ca.

Student Advisor: Kari Zacharias, Assistant Professor, Engineering Education, Price Faculty of Engineering, kari.zacharias@umanitoba.ca.

Conflicts of Interest and Undue Influence

Luke Slade does not have any conflicts of interest in this study. As a professor in the Price Faculty of Engineering, Kari Zacharias may be a past or future course instructor for some study participants. Dr. Zacharias is not involved in the recruitment process for

this study, and participating in this study will not affect your grades in any course or your standing at the University of Manitoba.

Invitation to Participate

You are being invited to participate in a research study. This consent form, a copy of which you can download or print, is only part of the process of informed consent. If you want more details about something mentioned here, or information not included here, feel free to ask Luke Slade or Kari Zacharias, whose contact information can be found above. Please take the time to read this document and any accompanying information carefully. It is very important that you understand:

- what is being asked of you,
- what the risks and benefits of participation are, and
- how the information you provide will be used and stored.

Purpose of the Study

I (Luke Slade) am doing this research for a thesis under the supervision of Kari Zacharias. This research study is about evaluating a developed academic course recommendation tool (a tool that recommends what courses to take in each semester) from the perspective of the targeted users of the tool (computer engineering students at the University of Manitoba). I am hoping to evaluate the tool's meaningfulness and receive feedback on changes that may increase the meaningfulness of the tool.

Study Procedures

If you decide to take part in the study, you will be interviewed online through Zoom. The interview should take about 1 hour to complete. The interview audio and video will be

recorded. When scheduling the interview, you will be required to provide the order in which you have completed your courses, grouped by semester.

Study Risks

This interview is minimal risk, meaning that the risks are no greater than what you might encounter in your daily life.

Study Benefits

There are no direct benefits to you by participating in this study, but we hope to learn more about the meaningfulness of academic course recommendation tools for computer engineering students so their values and perceptions can be incorporated to such tools intended for them.

Compensation

As a participant in this study, you will be given \$25 gift card of your choice from the University of Manitoba Book Store, Tim Hortons, or Starbucks to compensate you for your time participating in this study. This gift card will be delivered by email within 3 weeks from scheduling the interview.

Confidentiality

All the information you provide as a participant in this study is confidential which means we as researchers must keep it safe. We will do our best, however, it is not possible to guarantee absolute confidentiality. We will only share your personal information if required through court order or law.

Use and Storage of Information

The interview audio and video recording and audio transcript will be stored on Zoom's cloud storage through the University of Manitoba's license, only accessible to the SPI and advisor. The research team will be able to listen to the audio recording and read the transcript.

We may share some of your information as quotes. The quotes will not include your name or any information that may identify you.

We will destroy any identifiable information like your name by 05/2026. The anonymized transcripts will be destroyed by 06/2031.

Data Sharing

Your research information may be shared outside of the University of Manitoba with researchers, other organizations, and/or made publicly available. The information is being shared for further analysis or testing, as part of the research study, and/or because it is required by a funder or journal. It will not include your name or any information that could directly identify you.

Dissemination

The study results will be shared through

4. the publication of a master's thesis; and/or
5. journal publications and presentations in relevant academic fields; and/or
6. a summarized report of the study results to be sent to interested participants.

Withdrawing

Your participation in this research study is voluntary. You can choose to only answer the interview questions that you are comfortable with. You may withdraw from the study for any reason. To withdraw from the study, contact the SPI and indicate that you wish to withdraw from the study. You do not have to explain why. You will not be penalized in

any way. All your responses, the audio and video recordings, and transcript will be destroyed. If you wish to withdraw, you must do so before 14/04/2026.

Summary of Results

A summary of the study results will be available by 08/2026. If you wish to receive a copy by email, please indicate so to the SPI immediately after the interview, through email, or at the bottom of this consent form.

Questions or Concerns

Designated University of Manitoba personnel may check that this study is being done safely and properly. To do this, they may review the research records.

This study has been reviewed and approved by a Research Ethics Board at the University of Manitoba. However, this does not mean that participation is risk-free. If you have questions about your rights as a research participant, you may contact the Office of Human Research Ethics at humanethics@umanitoba.ca or (204) 474-7122.

If you have any questions, concerns, or complaints about this study, you may contact any members of the research team listed on the first page or the Office of Human Research Ethics.

Consent:

By participating in this study, I agree that:

- I have read the above information.
- I have had the opportunity to ask and have answered all of my questions.
- I understand what is being asked of me.

- I will be taking part in a research study.
- I may freely stop or leave the research study activities at any time.
- My information may be shared outside the University of Manitoba.
- I do not waive my legal rights by participating in the study.

Notice Regarding Collection, Use, and Disclosure of Personal Information

Your personal information is being collected under the authority of *The University of Manitoba Act*. The University of Manitoba is committed to preserving your right to privacy. The information you provide will be used by the University to support our research. Your personal information will not be used or disclosed for other purposes, unless permitted by *The Freedom of Information and Protection of Privacy Act* or *The Personal Health Information Act*. If you have any questions about the collection of personal information:

Ph: 204-474-9462 or Email: fippa@umanitoba.ca

The University of Manitoba wants you to know that the information you share with our researchers will only be used in the ways explained in this form and as Manitoba privacy laws allow. If you have questions, you can call 204-474-9462 or Email: fippa@umanitoba.ca.

I agree to participate in this study. Name: _____ Signature: _____	☐ YES ☐ NO
I would like to receive a summary of the results of this study.	☐ YES ☐ NO

If yes, please provide an email address/contact information: _____	
--	--

Appendix C: Cronbach's Alpha for Survey

Each table presents the number of courses with a minimum number of survey responses in each range of Cronbach's alpha.

Table 42. Cronbach's alpha for utility

Min Responses	$\alpha \geq 0.9$	$\alpha < 0.9, \geq 0.8$	$\alpha < 0.8, \geq 0.7$	$\alpha < 0.7, \geq 0.6$	$\alpha < 0.6, \geq 0.5$	$\alpha < 0.5$
1	17	18	5	1	1	5
2	17	18	5	1	1	3
3	16	18	4	1	0	2
4	15	18	3	1	0	1
5	15	17	3	1	0	0

Table 43. Cronbach's alpha for cost

Min Responses	$\alpha \geq 0.9$	$\alpha < 0.9, \geq 0.8$	$\alpha < 0.8, \geq 0.7$	$\alpha < 0.7, \geq 0.6$	$\alpha < 0.6, \geq 0.5$	$\alpha < 0.5$
1	17	15	9	2	2	2
2	17	15	9	2	2	0
3	14	15	8	2	2	0
4	13	14	8	1	2	0
5	12	13	8	1	2	0

Appendix D: Synthetic Course Pathways for Dataset

Table 44. Varying width simulated pathways

Graduation Year	4	4.5	5	5.5	6	6.5	7
Pathways	0	19	327,534	154,215	450,582	185	335
Percentage	0	0.00	35.1	16.5	48.3	0.02	0.04

Table 45. Static width simulated pathways

Graduation Year	4	4.5	5	5.5	6	6.5	7
Pathways	484	517	830,803	77,325	104,144	93,588	89,063
Percentage	0.04	0.04	69.5	6.47	8.71	7.83	7.45

Table 46. Combined simulated pathways

Graduation Year	4	4.5	5	5.5	6	6.5	7
Pathways	484	536	1,158,337	231,540	554,726	93,773	89,398
Percentage	0.02	0.03	54.4	10.9	26.1	4.40	4.20

Appendix E: Hyperparameter Optimization Results and Training Behaviour

This appendix contains my hyperparameter optimization (HPO) results, of the Courses Taken Vector (CTV), Semester Sequence, and Hybrid models. This portion of the implementation process, as displayed in Figure 31, consisted of two rounds of HPO for each model type (with the exception of the CTV model), and resulted in candidate models based on the highest validation accuracy from each round of HPO.

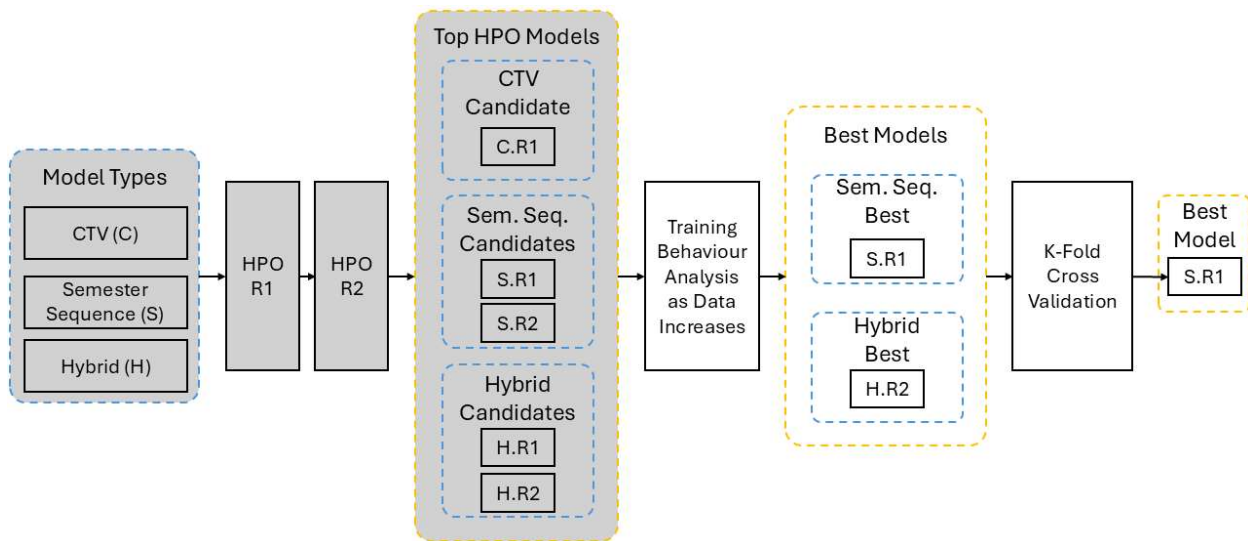


Figure 31. Hyperparameter optimization portion of the implementation process.

CTV Broad Search HPO Results

The top 3 most accurate configurations of the CTV model based on the recall@6 of the validation set are displayed in Table 47 alongside the other performance metrics of AUC (ranking ability) and KLD (probability distribution representation).

Table 47. Top 3 CTV HPO models from the broad search (round 1)

Metric	Top Trial 1	Top Trial 2	Top Trial 3
Validation Recall@6	0.7078	0.6957	0.6556
Validation AUC	0.1821	0.1812	0.1816
Validation KLD	13.40	13.42	13.40
Parameter			
Learning Rate	1.771e-4	2.810e-4	2.643e-4
KLD Weight	0.4880	0.9611	0.3813
Embedding	512	512	512
Layers	4	3	3
Heads per Layer	16	16	16
Batch Size	128	128	128

The continuous parameters, learning rate and KLD weight, converged to respective magnitudes of $1e-4$ and $1e-1$ respectively. The discrete parameters also converged to a single number or 2 close numbers in the case of the number of layers. Figure 32 displays the search space coverage for each parameter against model accuracy.

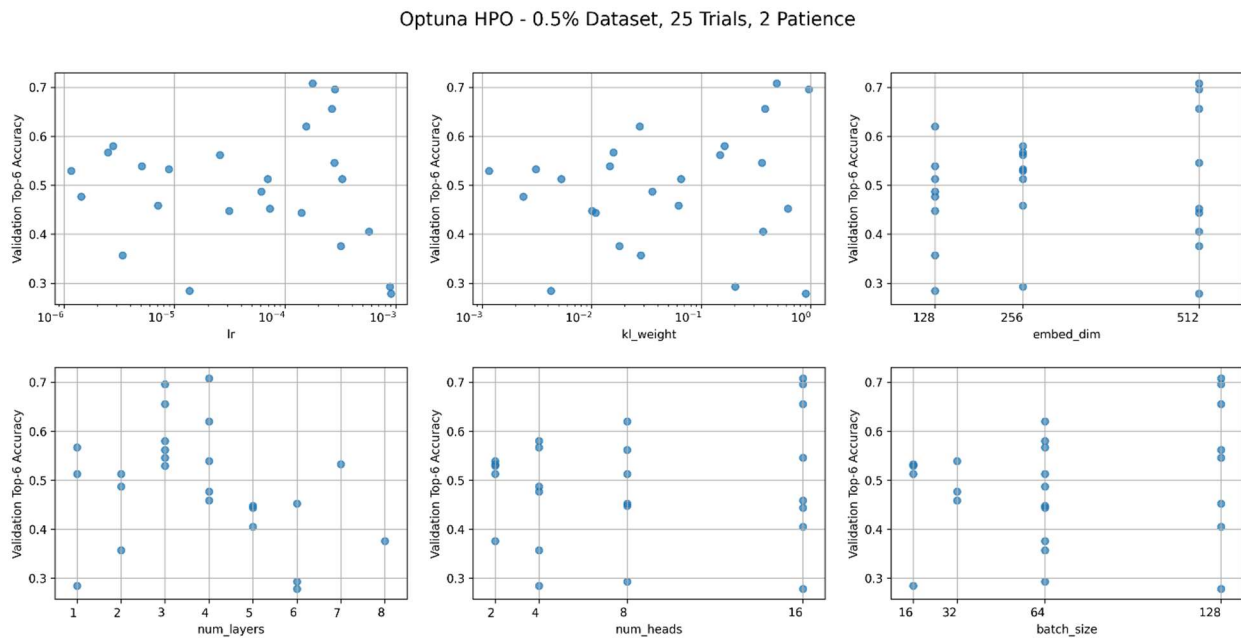


Figure 32. CTV model HPO round 1 search space coverage and accuracy

The search space appeared to be sufficiently searched as each discrete hyperparameter had been tested at least once and the continuous hyperparameters had been tested throughout the range of possible numbers. The highest accuracy hyperparameters tended to be somewhat clustered together, suggesting that Optuna converged towards an optimal configuration. To view the convergence, Figure 33 plots the highest accuracy achieved during HPO throughout the 25 trials.

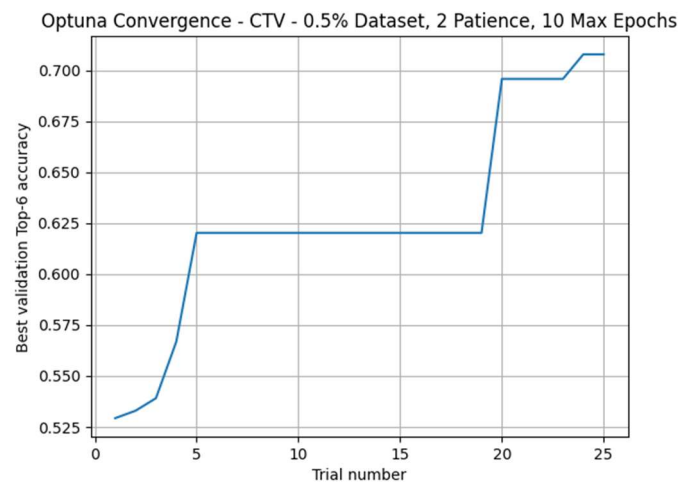


Figure 33. CTV HPO round 1 accuracy convergence

As the highest accuracy of the 25 trials occurred on trial 24, further trials may have further increased accuracy. However, when compared to the accuracies of the Semester Sequence and Hybrid models over 25 trials the CTV model has somewhat poor performance, especially regarding the metric of AUC that suggests that the CTV model struggled to learn to rank positive outputs above negative outputs. As such, only the first round of HPO for the CTV model was conducted as refining the search space would likely not result in a large increase in accuracy or ranking.

Semester Sequence Broad Search HPO Results

The top 3 most accurate configurations of the Semester Sequence model, after the broad search HPO are displayed in Table 48 alongside their performance metrics.

Table 48. Top 3 Semester Sequence HPO models from the broad search (round 1)

Metric	Top Trial 1	Top Trial 2	Top Trial 3
Validation Recall@6	0.9187	0.9178	0.9178
Validation AUC	0.9686	0.9607	0.9662
Validation KLD	9.387	9.706	9.500
Parameter			
Learning Rate	3.147e-4	8.010e-5	7.716e-5
KLD Weight	0.7900	0.05635	0.2539
Embedding	512	512	512
Layers	6	4	7
Heads per Layer	4	4	4
Batch Size	64	16	16

Within the top 3 most accurate models, the most accurate is top trial 1 with a validation recall@6 of 0.9187, the best ranking model was also top trial 1 with a validation AUC of 0.9686, and the best probability distribution modelling was top trial 1 with a validation KLD of 9.387. The top trials 2 and 3 both had comparable validation accuracy, ranking, and probability modelling as top trial 1. Some parameter convergence occurred as seen in the top 3 model in Table 48 and the clusters of high accuracy hyperparameter values in Figure 34.

Optuna HPO - 0.5% Dataset, 25 Trials, 2 Patience

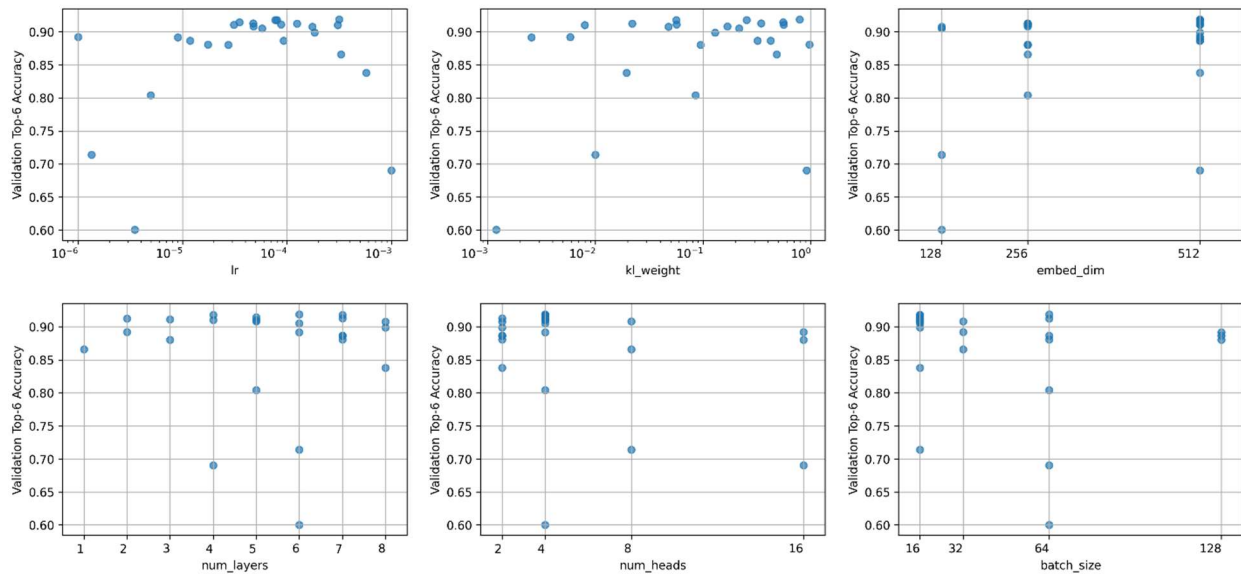


Figure 34. Semester Sequence model HPO search space coverage and accuracy

The most accurate learning rates were clustered between magnitudes of $1e-5$ and $1e-4$, the most accurate KLD weights were broadly between $1e-2$ and 1, performance was slightly higher with an embedding dimension of 512 compared to 256, the models with 4 to 6 layers had the highest accuracies, with 2 to 4 attention heads in each layer being the most accurate, and batch sizes of 64 and 16 being the most accurate. The search space had sufficient coverage with each discrete value being tested, values of continuous hyperparameters occurring throughout the total available range, and the emergent accuracy trends previously mentioned. Figure 35 displays the convergence of the highest accuracy model during HPO across 25 trials.

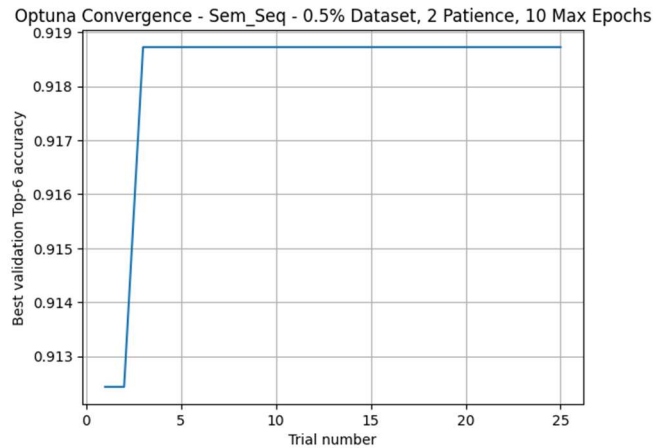


Figure 35. Semester Sequence HPO round 1 accuracy convergence

The highest accuracy configuration was found in the third trial, with no improvements made in the remaining 22 trials. With 22 trials being performed with any accuracy improvements, further trials would likely not result in the discovery of better hyperparameter configurations.

Semester Sequence Refined Search HPO Results

The top 3 most accurate configurations of the Semester Sequence model from the refined search HPO (round 2) are displayed in Table 49 alongside their performance metrics.

Table 49. Top 3 Semester Sequence HPO models from the refined search (round 2)

Metric	Top Trial 1	Top Trial 2	Top Trial 3
Validation Recall@6	0.9367	0.9352	0.9223
Validation AUC	0.9483	0.9523	0.9623
Validation KLD	10.16	10.04	9.700
Parameter			
Learning Rate	2.230e-5	1.832e-5	3.400e-4
KLD Weight	4.576e-2	5.790e-2	2.083e-1
Embedding	1024	1024	1024
Layers	7	7	7
Heads per Layer	4	4	4
Batch Size	32	32	32

Within the top 3 most accurate models, the most accurate is top trial 1 with a validation recall@6 of 0.9367, the best ranking model was top trial 3 with a validation AUC of 0.9623, and the best probability distribution modelling was also top trial 3 with a validation KLD of 9.700. All 3 top trials had comparable validation accuracy, ranking, and probability distribution modelling. Much convergence of hyperparameters occurred in the refined search as seen in the configurations in Table 49 and in the coverage of the search space in Figure 36.

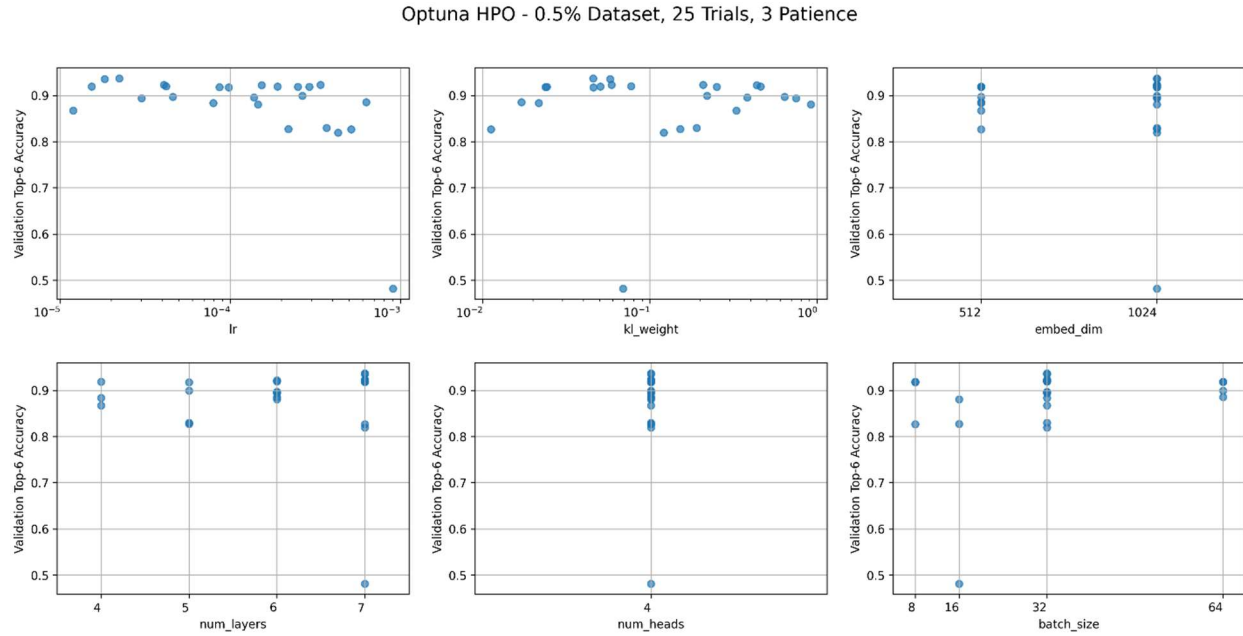


Figure 36. Semester Sequence model HPO round 2 search space coverage and accuracy

The most accurate models were found to have the following, learning rates clustered between magnitudes of $1e-5$ and $1e-4$, KLD weights between $1e-2$ and $1e-1$, embedding dimension of 1024, 7 Transformer layers, and a batch size of 32. The number of attention heads was held constant for this round of HPO. The search space had sufficient coverage with each discrete value being tested, continuous values spaced throughout the total range, and the emergent accuracy trends previously mentioned. Figure 37 displays the convergence of the highest accuracy during HPO across the 25 trials.

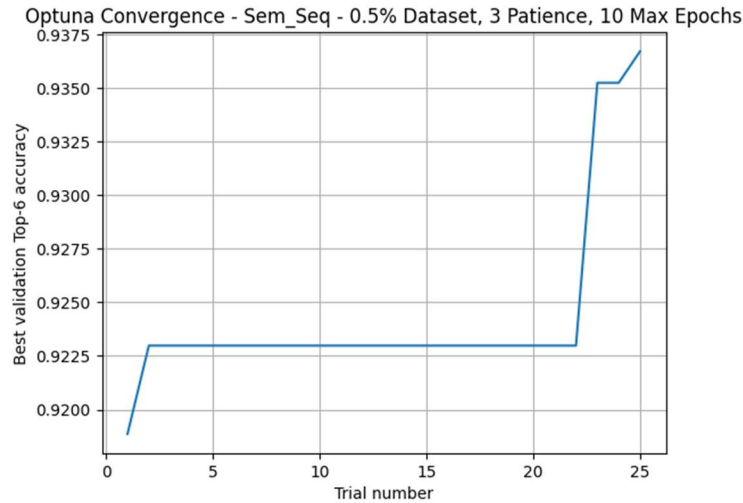


Figure 37. Semester Sequence HPO round 2 accuracy convergence

The highest accuracy configuration was found in the final trial (25), and the second most accurate configuration was found on trial 23. These late discoveries suggest that further trials would likely lead to the discovering more accurate configurations, however, the increase in accuracy across the 25 trials was approximately 0.018, which suggests that further trials may not lead to much improvement. As the top trials are already very accurate, 25 trials was sufficient given the scope of my research. The change in performance metrics of the top 3 HPO trials from the broad search (HPO round 1) and the refined search (HPO round 2) are shown in Table 50.

Table 50. Change in performance metrics of top 3 Semester Sequence models from HPO round 1 to round 2

Change in Metric	Top Trial 1	Top Trial 2	Top Trial 3
Validation Recall@6	+ 0.01797	+ 0.01741	+ 0.005204
Validation AUC	- 0.02028	- 0.008476	- 0.003886
Validation KLD	+ 0.7773	+ 0.3350	+ 0.1999

All 3 of the top trials from the refined search had accuracy improvements, while AUC and KLD worsened. These results suggest that as the accuracy increases, AUC decreases and KLD increases, however it is not inherently proportional as both top trial 1 and top trial 2 had similar accuracy increases but the changes to AUC and KLD were not similar. The slight improvement in accuracy but worsening in AUC and KLD of each of the top trials highlights the need for further investigation and consideration before selecting the best Semester Sequence model.

Hybrid Broad Search HPO Results

The top 3 most accurate configurations of the Hybrid model, after the broad search HPO are displayed in Table 51 alongside their performance metrics.

Table 51. Top 3 Hybrid HPO models from the broad search (round 1)

Metric	Top Trial 1	Top Trial 2	Top Trial 3
Validation Recall@6	0.9346	0.9216	0.9173
Validation AUC	0.8673	0.9255	0.9219
Validation KLD	13.09	10.81	10.97
Parameter			
Learning Rate	2.882e-4	1.744e-4	1.946e-4
KLD Weight	1.145e-3	5.418e-3	5.457e-3
Embedding	512	512	512
Layers	5	7	7
Heads per Layer	16	2	2
Batch Size	32	16	16

Within the top 3 most accurate models, the most accurate is top trial 1 with a validation recall@6 of 0.9346, the best ranking model was top trial 2 with a validation AUC of 0.9255, and the best

probability distribution modelling was also top trial 2 with a validation KLD of 10.81. Top trial 1's accuracy was much better than top trials 2 and 3 whereas top trials 2 and 3 had comparable accuracies. However, top trial 1 had worse ranking ability than top trials 2 and 3, and top trial 1 had worse probability distribution modelling than top trials 2 and 3. Many parameters converged as seen in Table 51 and the coverage displayed in Figure 38.

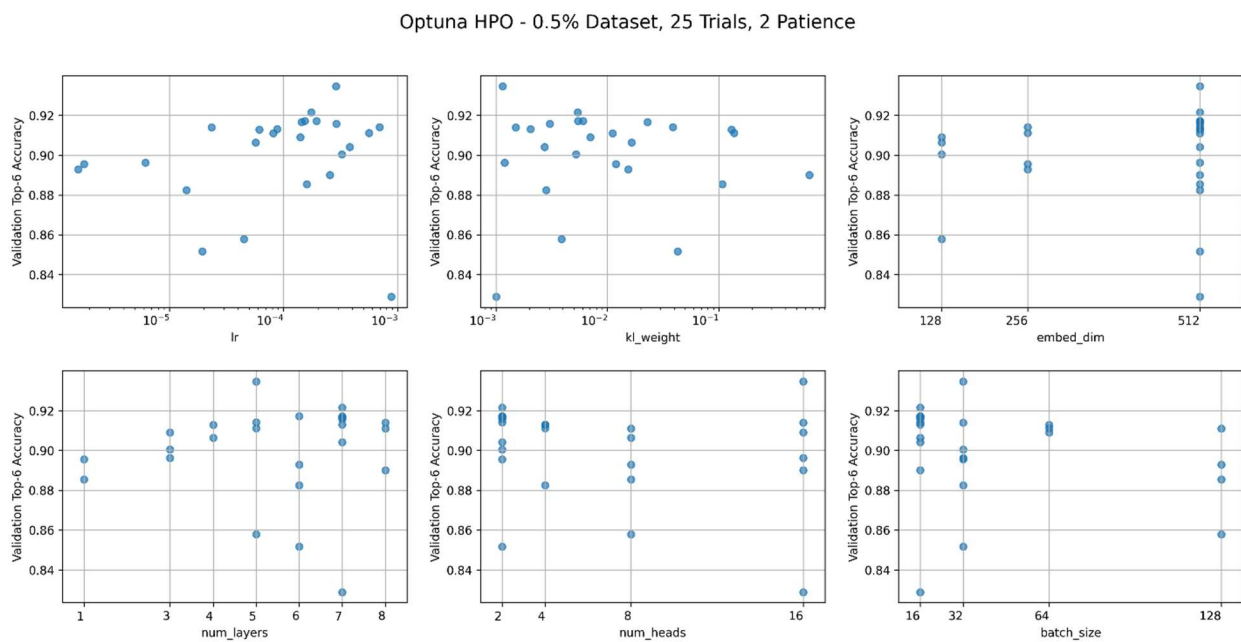


Figure 38. Hybrid model HPO round 1 search space coverage and accuracy

The most accurate models were found to have the following, learning rates in the magnitude of $1e-4$, KLD weights in the magnitude of $1e-3$, embedding dimension of 512, 5 or 7 Transformer layers, with 2 or 16 attention heads per layer, and a batch size of 32 or 16. The search space had sufficient coverage with each discrete value being tested, continuous values spaced throughout the total range, and the emergent accuracy trends previously mentioned. Figure 39 displays the convergence of the highest accuracy trials during HPO.

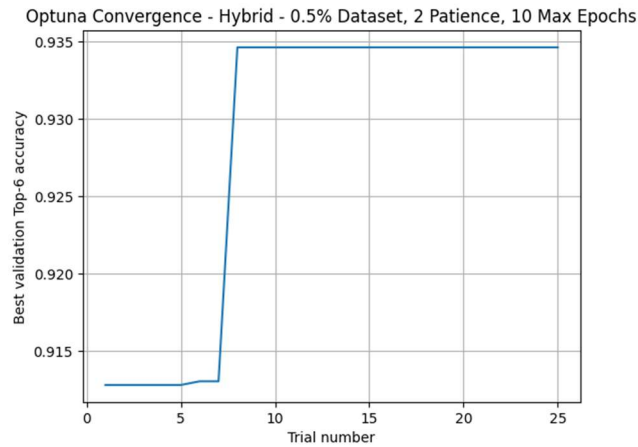


Figure 39. Hybrid HPO round 1 accuracy convergence

In terms of performance convergence, the highest accuracy configuration was found in the 8th trial with no further improvements in the remaining 17 trials. This suggests that more trials would likely not result in the discovery of more accurate configurations. Through this round of HPO there was an approximate increase in accuracy of 0.02 from the first trial to the most accurate trial.

Hybrid Refined Search HPO Results

The top 3 most accurate configurations of the Hybrid model, after the refined search HPO are displayed in Table 52 alongside their performance metrics.

Table 52. Top 3 Hybrid HPO models from the refined search (round 2)

Metric	Top Trial 1	Top Trial 2	Top Trial 3
--------	-------------	-------------	-------------

Validation Recall@6	0.9224	0.9224	0.9223
Validation AUC	0.9153	0.9145	0.9206
Validation KLD	11.21	11.20	10.96
Parameter			
Learning Rate	2.167e-4	1.358e-4	1.785e-4
KLD Weight	4.682e-3	4.635e-3	5.435e-3
Embedding	512	512	512
Layers	5	5	5
Heads per Layer	8	16	8
Batch Size	16	16	16

Within the top 3 most accurate models, the most accurate is top trial 1 with a validation recall@6 of 0.9224, the best ranking model was top trial 3 with a validation AUC of 0.9206, and the best probability distribution modelling was also top trial 3 with a validation KLD of 10.96. All three of the top trials resulted in comparable results for accuracy, ranking ability, and probability distribution modelling. Many parameters converged as seen in Table 52 and Figure 40. The most accurate models were found to have the following, learning rates between 1.2e-4 and 2.4e-4, KLD weights between 4e-3 and 6e-3, 5 Transformer layers, with 8 or 16 attention heads per layer, and a batch size of 16. Figure 40 displays the coverage of the refined search space.

Optuna HPO - 0.5% Dataset, 25 Trials, 3 Patience

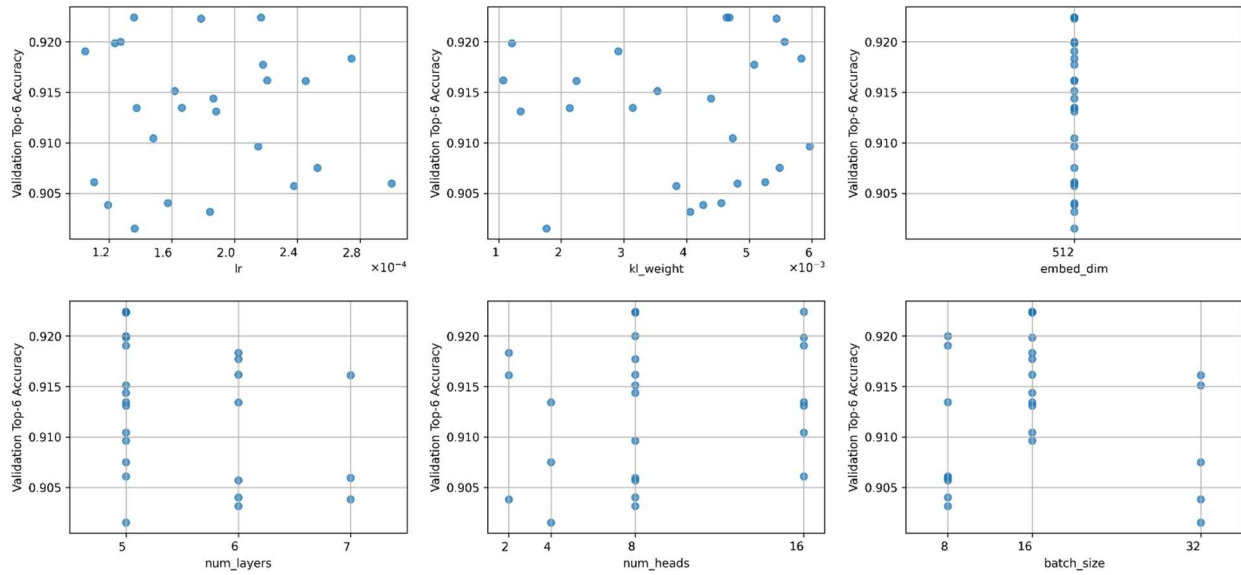


Figure 40. Hybrid model HPO round 2 search space coverage and accuracy

The search space had sufficient coverage with each discrete value being tested, continuous values spaced throughout the total range, and the emergent accuracy trends previously mentioned. Figure 41 displays the convergence of the highest accuracy during HPO.

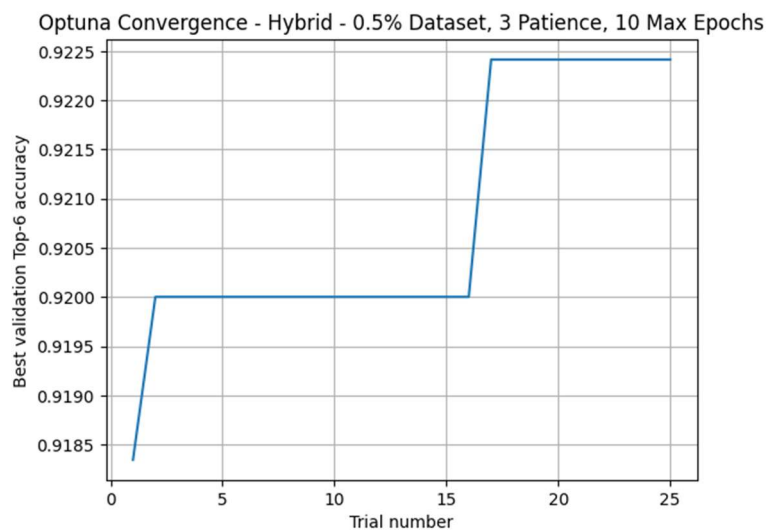


Figure 41. Hybrid HPO round 2 accuracy convergence

The highest accuracy configuration was found in the 17th trial with no further improvements in the remaining 8 trials. This suggests that more trials would likely not result in the discovery of more accurate configurations. Through this round of HPO, there was an approximate increase in accuracy of 0.004 from the first trial to the most accurate trial, which suggests that further trials may not have any significant improvements to accuracy. The change in performance metrics from the Hybrid model top 3 trials of the broad search to the top 3 trials of the refined search are shown in Table 53.

Table 53. Change in performance metrics of top 3 Hybrid models from round 1 to round 2

Change in Metric	Top Trial 1	Top Trial 2	Top Trial 3
Validation Recall@6	- 0.01221	+ 0.0008040	+ 0.005062
Validation AUC	+ 0.04802	- 0.01094	- 0.001284
Validation KLD	- 1.878	+ 0.3865	- 0.009492

The highest accuracy configuration showed a slight decrease in accuracy and a slight improvement to AUC alongside an improvement to KLD. Top trials 2 and 3 showed slight improvements to accuracy and slightly worse AUC. Top trial 2 showed worse KLD while top trial 3 saw a slight improvement to KLD. The decrease in accuracy alongside the improvements to AUC and KLD seen in top trial 1 highlighted the need to further investigate both the round 1 and round 2 models before selecting the better model.

Expanding Training Dataset Size Results

To see how the performance of each model changes as the size of the dataset increases, I trained and compared the candidate models of each input type using 1%, 2%, and 5% of the total available training data, this part of the model selection process is highlighted in Figure 42 (refer back to Table 26 for dataset sizes).

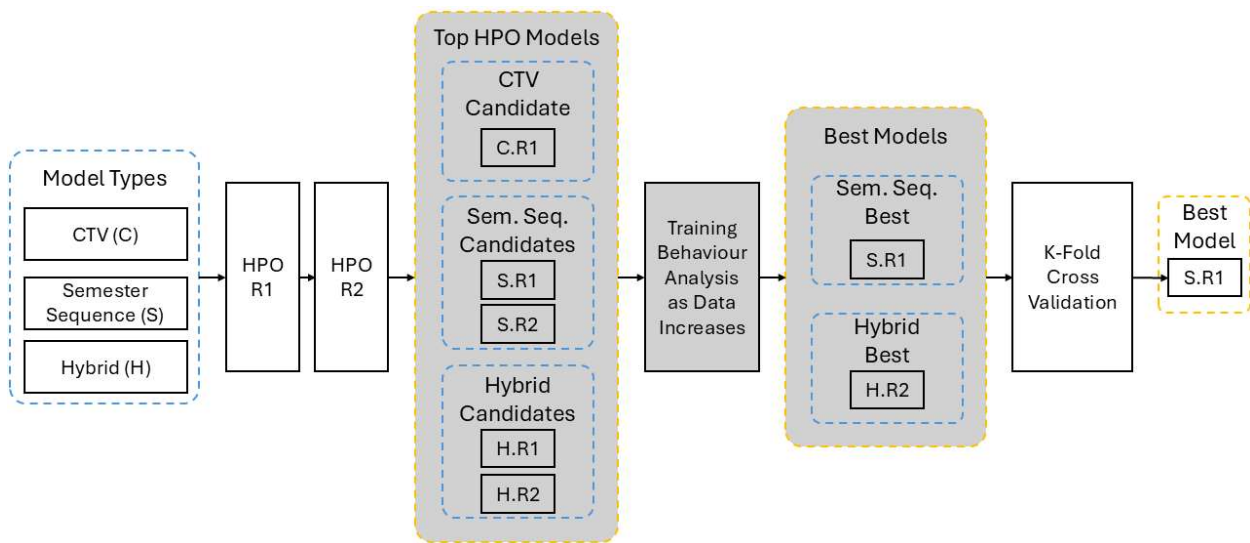


Figure 42. Training behaviour analysis portion of the model selection process

The top HPO model for the CTV model was investigated using 1% of the training dataset to see if improvements were made from the HPO that used 0.5% of the training dataset. As performance was insufficient, investigation into the CTV model was concluded. The Semester Sequence and Hybrid models showed strong and comparable performance. Both the top models from round 1 (broad search) and round 2 (refined search) were compared on 1% and 2% of the training data to determine which model has higher performance as the dataset increases. As highlighted in Figure 42, the resulting best models from the training behaviour analyses were the

configuration from HPO round 1 for the Semester Sequence model and the configuration from HPO round 2 for the Hybrid model.

CTV Expanding Training Dataset Size Results

The results of the CTV model from HPO were unsatisfactory (recall@6 of 0.708, AUC of 0.182, and KLD of 13.4), however the training performance of the model when the dataset size was investigated as it increased from 0.5% during HPO to 1% to confirm its lack of performance when training on a larger dataset. Although there was a slight increase in validation accuracy during the first 3 epochs on the 1% dataset compared to the 0.5% dataset, the validation ranking performance did not improve during those epochs. The highest ability to rank was seen at epoch 16 with an AUC of 0.1825 at which point the validation accuracy was 0.554. This performance analysis further revealed that the CTV model was much worse than the Semester Sequence and Hybrid models.

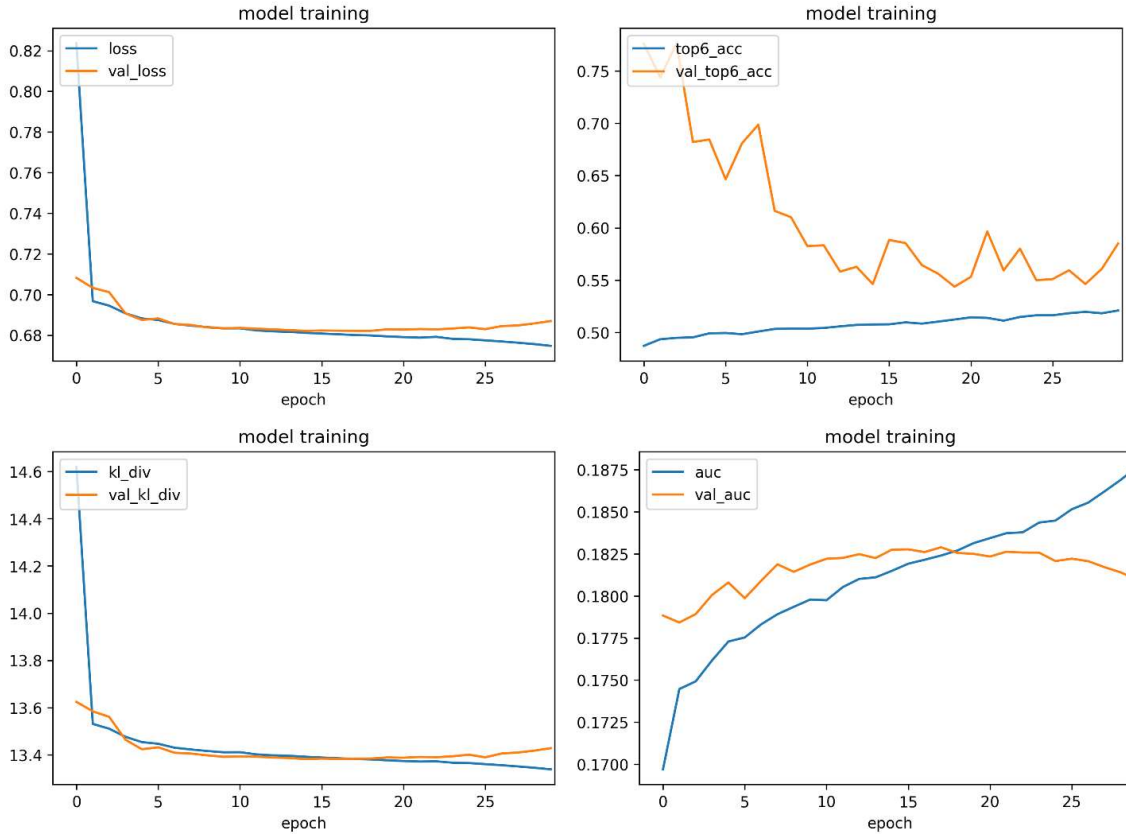


Figure 43. Optimized CTV model training behaviour on 1% of dataset for 30 epochs

Semester Sequence Model Selection Results

The hyperparameter configurations of the top model from the 2 rounds of HPO are shown in Table 54, and their metric results are shown in Table 55.

Table 54. Configurations of top HPO Semester Sequence models from each round of HPO

HPO Configuration Results – Semester Sequence		
Hyperparameter	Round 1	Round 2
Learning rate	3.147e-4	2.230e-05
KLD weight	0.7900	0.04576
Embedding dimension	512	1024
Layers	6	7
Heads per layer	4	4
Batch size	64	32

Table 55. Metric results of top HPO Semester Sequence models from each round of HPO

HPO Metric Results – Semester Sequence		
Metric	Round 1	Round 2
Validation Recall@6	0.9187	0.9367
Validation AUC	0.9686	0.9483
Validation KL Divergence	9.387	10.16

The training behaviour of the 2 models were compared on 1% and 2% of the training dataset to identify which model has the best generalization for new data. The top R1 model appeared to have better generalization for recall@6, KLD, and AUC. The top R2 model appeared to lose its ability to generalize at a faster rate than the R1 model as epochs increased. Across datasets, the models trained on the 2% dataset were slightly more generalized than the models trained on the 1% dataset. As my CRT was meant to be trained on a larger portion of the total dataset of values-oriented course pathways, the top R1 model was better suited than the top R2 model based on ability to generalize.

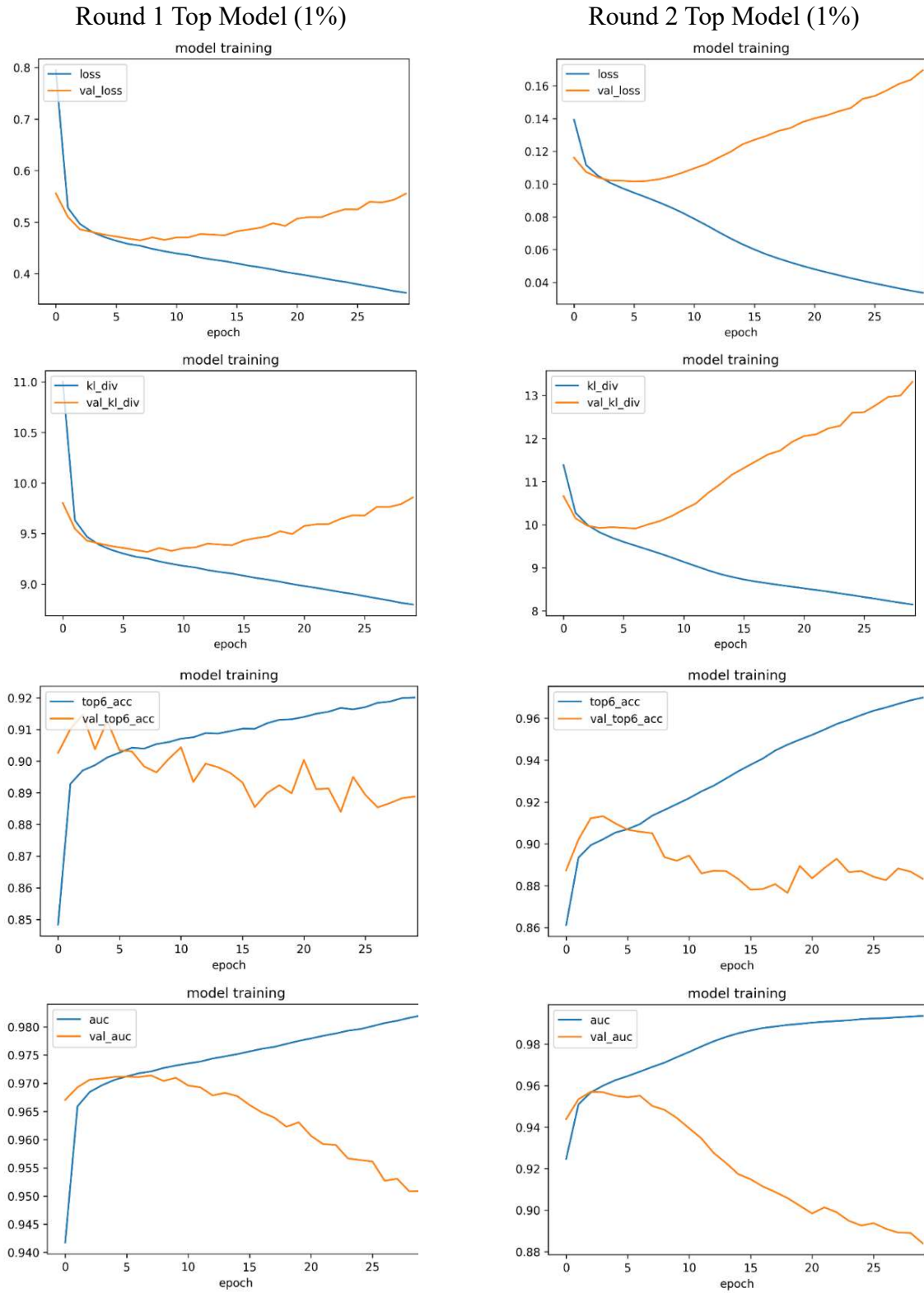


Figure 44. Comparison of training behaviour of the top Semester Sequence HPO models (round 1 left, round 2 right) on 1% of the dataset

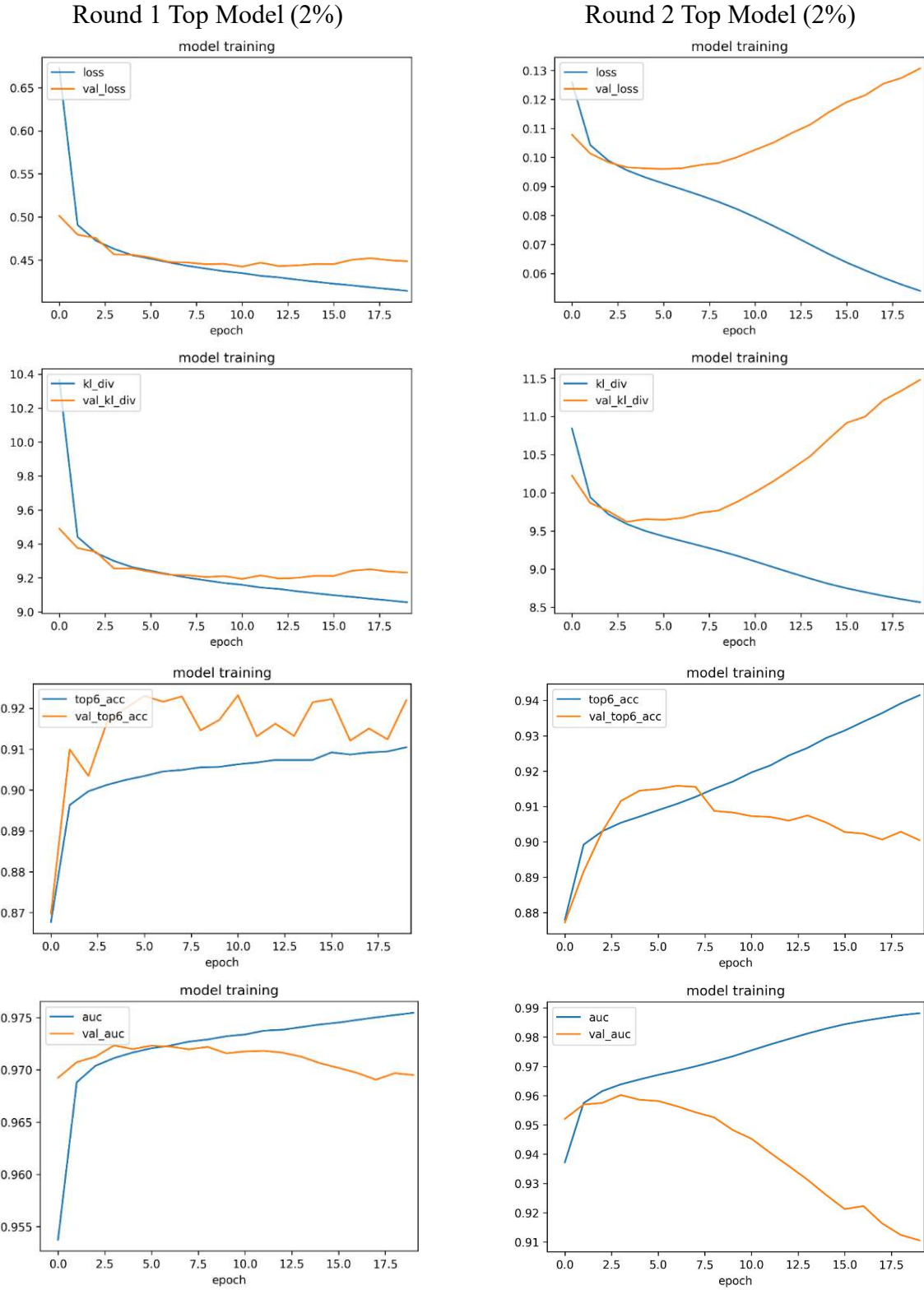


Figure 45. Comparison of training behaviour of the top Semester Sequence HPO models (round 1 left, round 2 right) on 2% of the dataset

Hybrid Model Selection Results

The hyperparameter configurations of the top Hybrid models from each round of HPO are shown in Table 56 with performance metrics reported in Table 57.

Table 56. Hyperparameter configurations of top HPO hybrid models from each round

HPO Configuration Results – Hybrid		
Hyperparameter	Round 1	Round 2
Learning rate	2.882e-4	2.167e-4
KLD weight	1.145e-3	4.682e-3
Embedding dimension	512	512
Number of layers	5	5
Number of heads per layer	16	8
Batch size	32	16

Table 57. Performance metrics of top HPO hybrid models from each round of HPO

HPO Metric Results – Hybrid		
Metric	Round 1	Round 2
Validation Recall@6	0.9346	0.9224
Validation AUC	0.8673	0.9153
Validation KL Divergence	13.09	11.21

The training behaviour of the 2 models were compared on 1% and 2% of the training dataset to identify which model has the best generalization for new data. On the same 2% dataset, the top R2 model appeared to have better generalization in accuracy (recall@6), modelling of the output probability distribution (KLD), ranking (AUC), and was more stable in higher epochs compared to the top round 1 model. Across datasets, the R1 model was slightly better when trained on the 2% dataset compared to its 1% results. Whereas the R2 model had slightly higher accuracy on

the 2% dataset than on the 1%. Since the R2 model had higher performance than the R1 model, the R2 model was selected as the best Hybrid model.

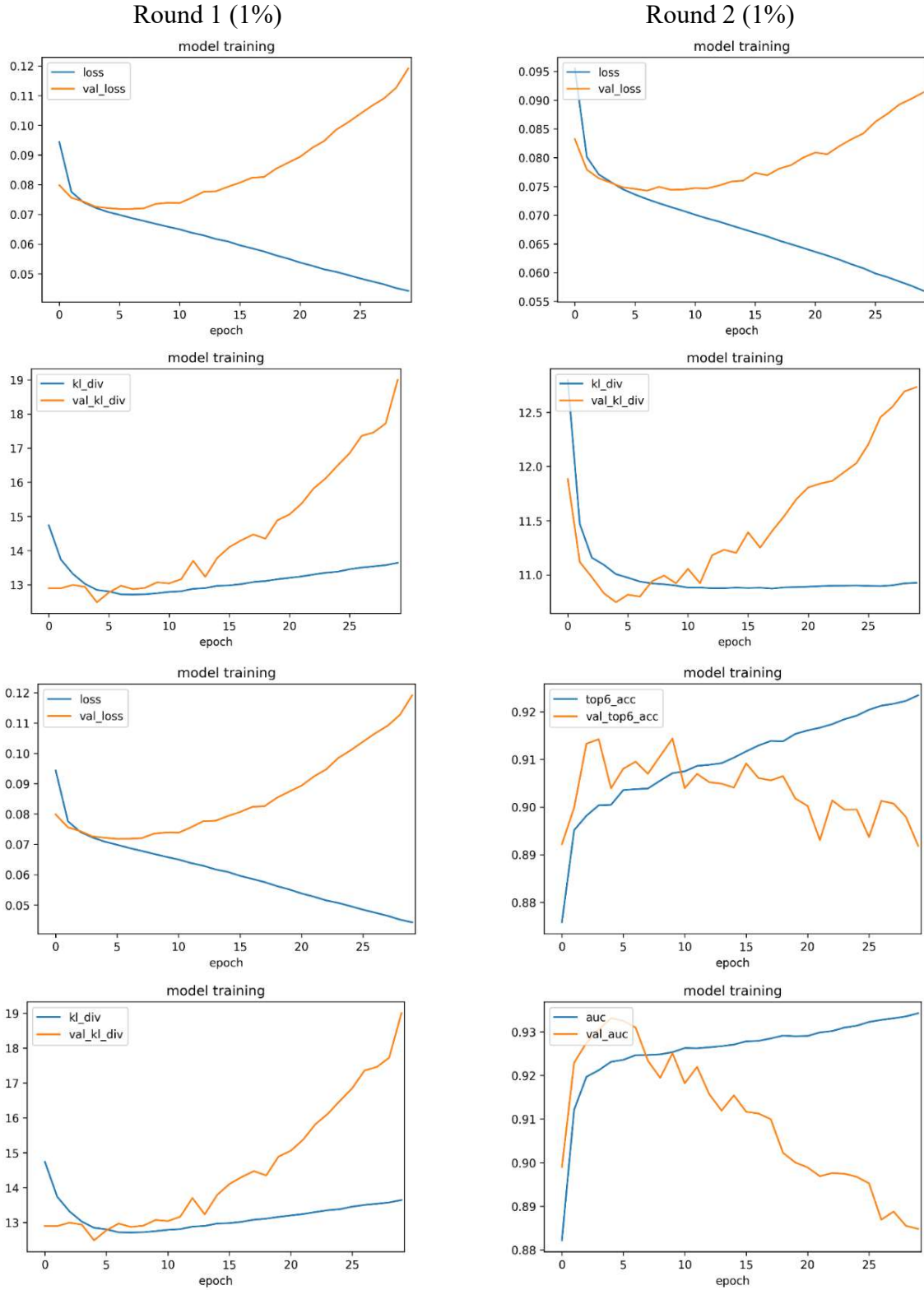


Figure 46. Comparison of training behaviour of the top Hybrid HPO models (round 1 left, round 2 right) on 1% of the dataset



Figure 47. Comparison of training behaviour of the top Hybrid HPO models (round 1 left, round 2 right) on 2% of the dataset

*HPO Refined Search Parameters**Table 58. Refined search space for the Semester Sequence model*

Parameter	Min	Max	Step Type (step size)
Learning rate	10^{-5}	10^{-3}	Continuous
KLD weight	10^{-2}	10^0	Continuous
Embedding dimension	512	1024	Discrete (2^n)
Transformer layers	4	7	Discrete (1)
Transformer heads per layer	4	4	-
Batch size	8	64	Discrete (2^n)

Table 59. Refined search space for the Hybrid model

Parameter	Min	Max	Step Type (step size)
Learning rate	1×10^{-4}	3×10^{-4}	Continuous
KLD weight	1×10^{-3}	6×10^{-3}	Continuous
Embedding dimension	512	512	-
Transformer layers	5	7	Discrete (1)
Transformer heads per layer	2	16	Discrete (2^n)
Batch size	8	32	Discrete (2^n)

*Timing Tables**Table 60. HPO round 1 timing*

HPO Round 1 (max epochs: 10, trials: 25, patience: 2)	
Model	Time (minutes)
CTV	149
Semester Sequence	326
Hybrid	472

Table 61. HPO round 2 timing

HPO Round 2 (max epochs: 10, trials: 25, patience: 3)	
Model	Time (minutes)
Semester Sequence	1054
Hybrid	478

Table 62. Training times of best models

Best Models			
Model	Dataset	Epochs	Time (minutes)
CTV	1%	30	92
Semester Sequence R1	1%	30	106
Semester Sequence R2	1%	30	527
Hybrid R1	1%	30	317
Hybrid R2	1%	30	228
Semester Sequence R1	2%	20	156
Semester Sequence R2	2%	20	738
Hybrid R1	2%	20	433
Hybrid R2	2%	20	313
Semester Sequence R1	5%	10	182
Hybrid R2	5%	10	329

Table 63. 5-Fold cross validation timing

5-Fold Cross Validation (5% Dataset, 10 epochs)	
Model	Time (minutes)
Semester Sequence R1	1019
Hybrid R2	1887

Appendix F: Evaluation Sensitivity Plots

Sobol Plots

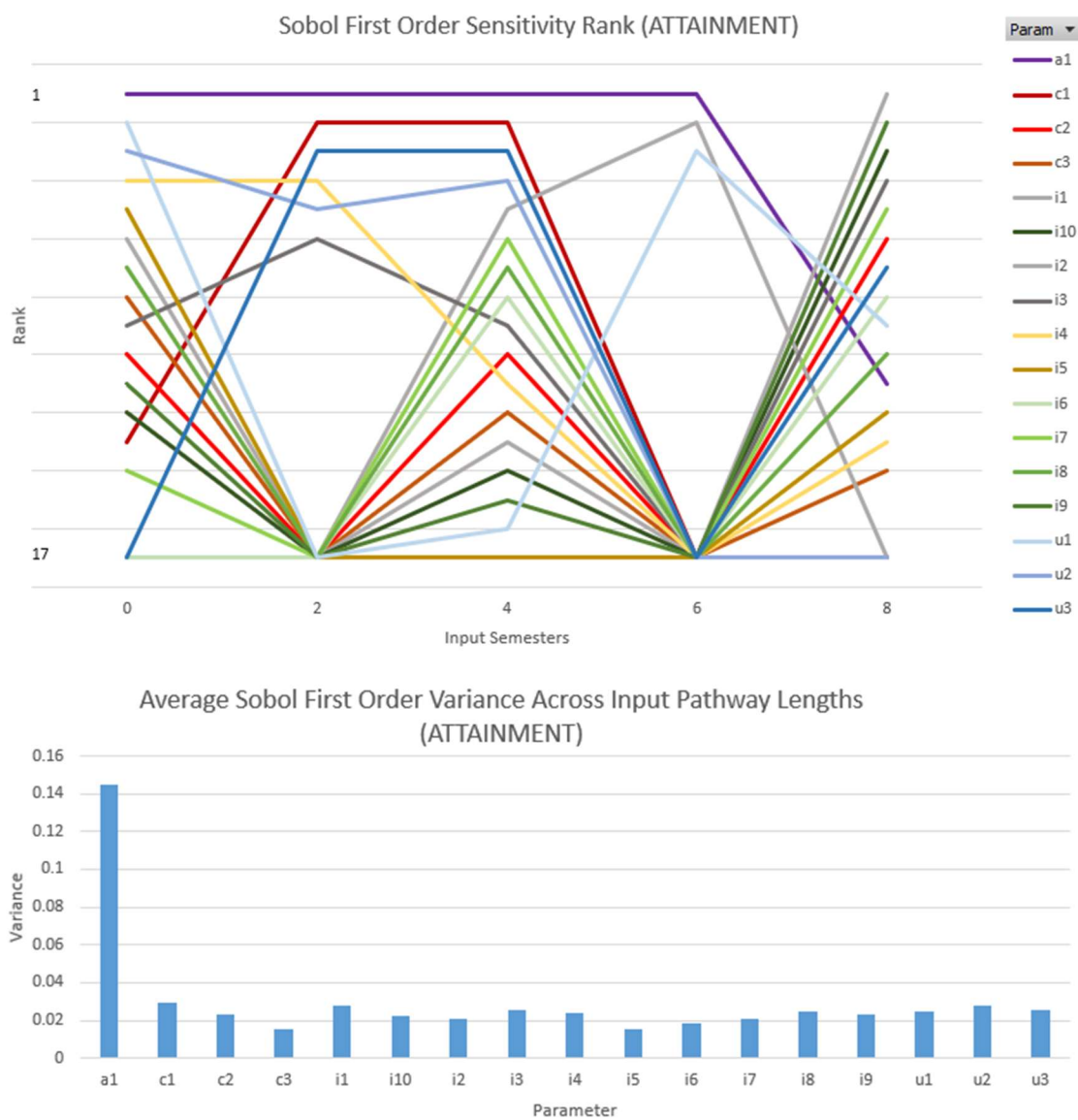


Figure 48. Sobol analysis attainment plots

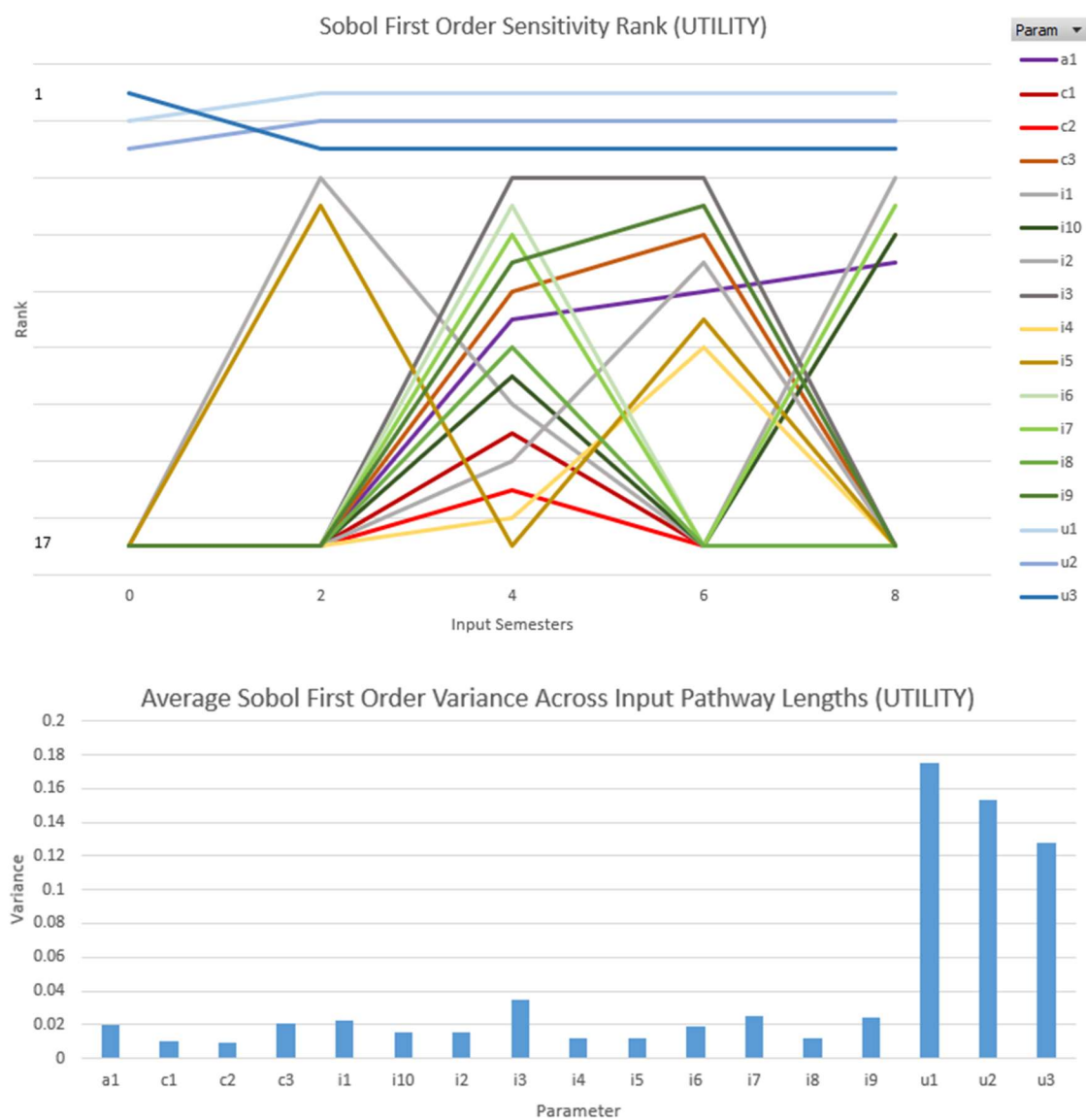


Figure 49. Sobol analysis utility plots

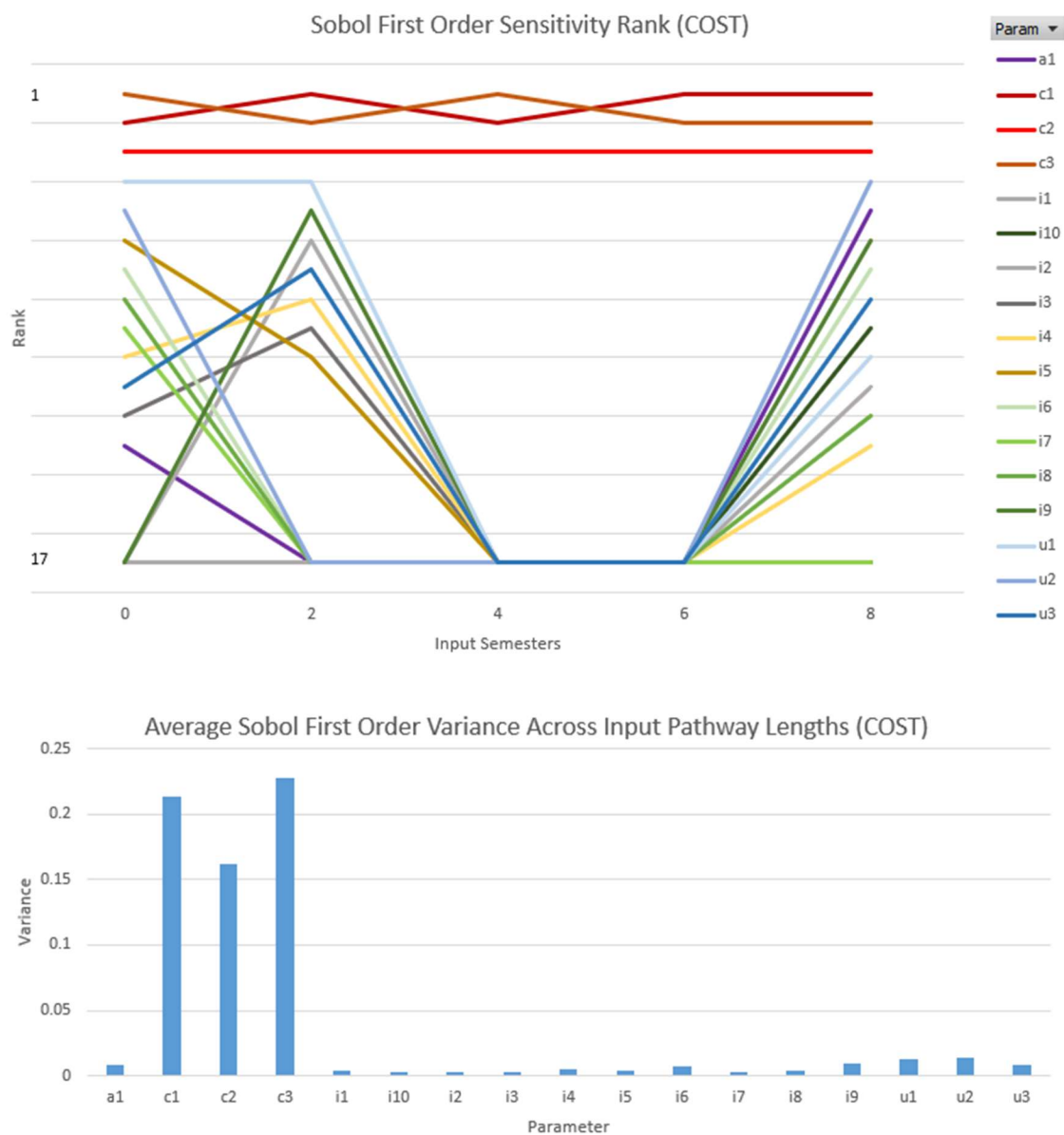


Figure 50. Sobol analysis cost plots



Figure 51. Sobol analysis interest plots

Delta Plots

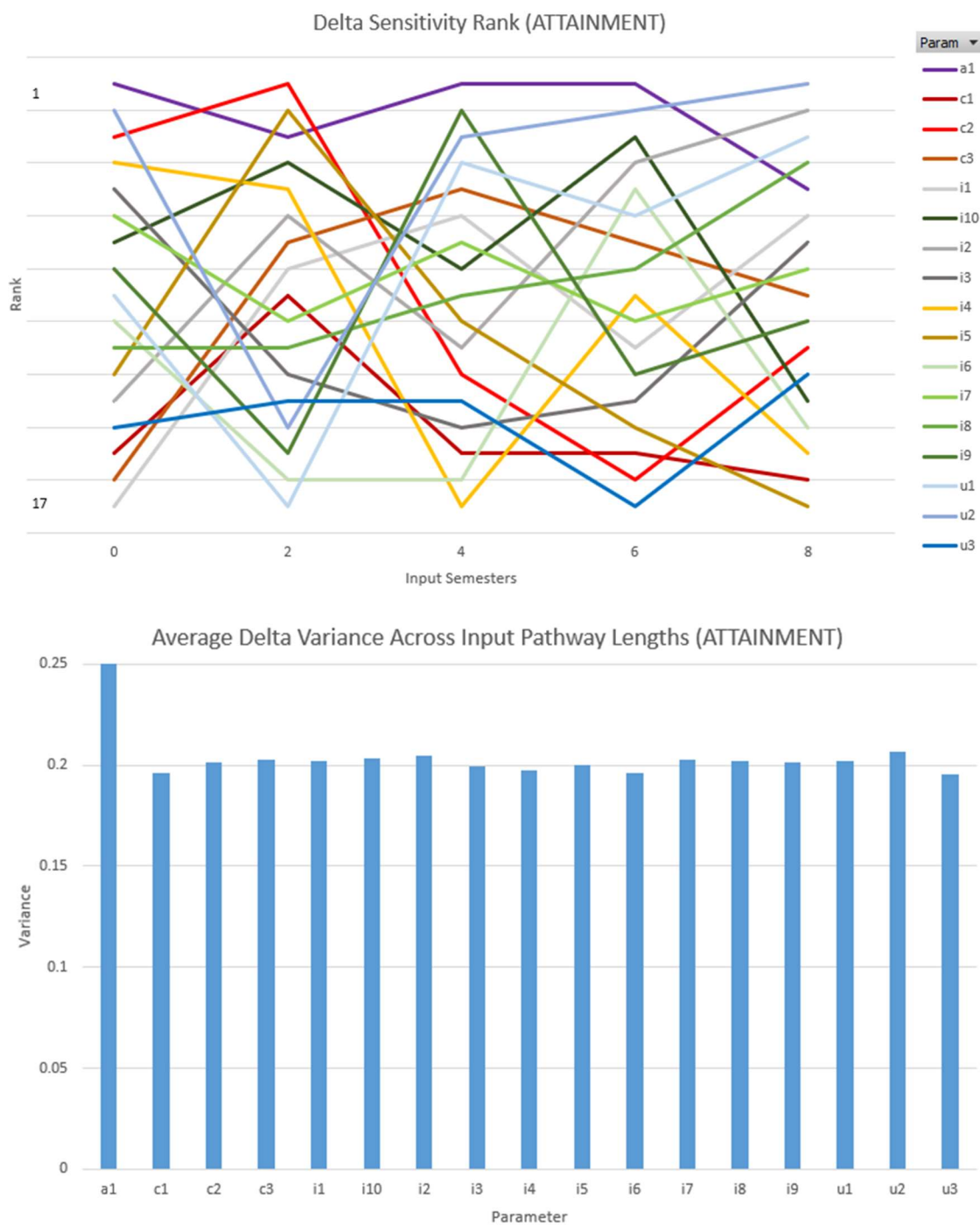


Figure 52. Delta analysis attainment plots

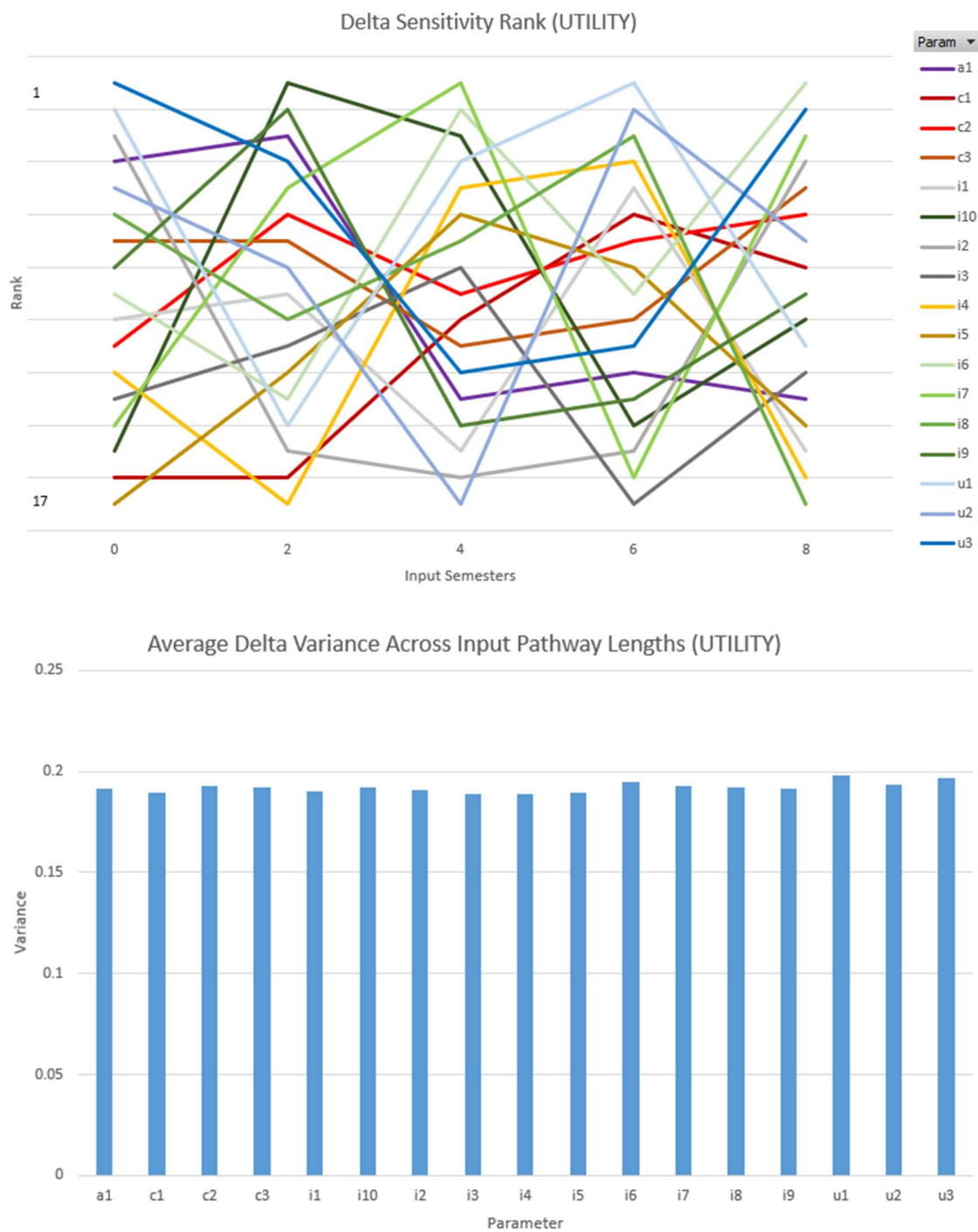


Figure 53. Delta analysis utility plots

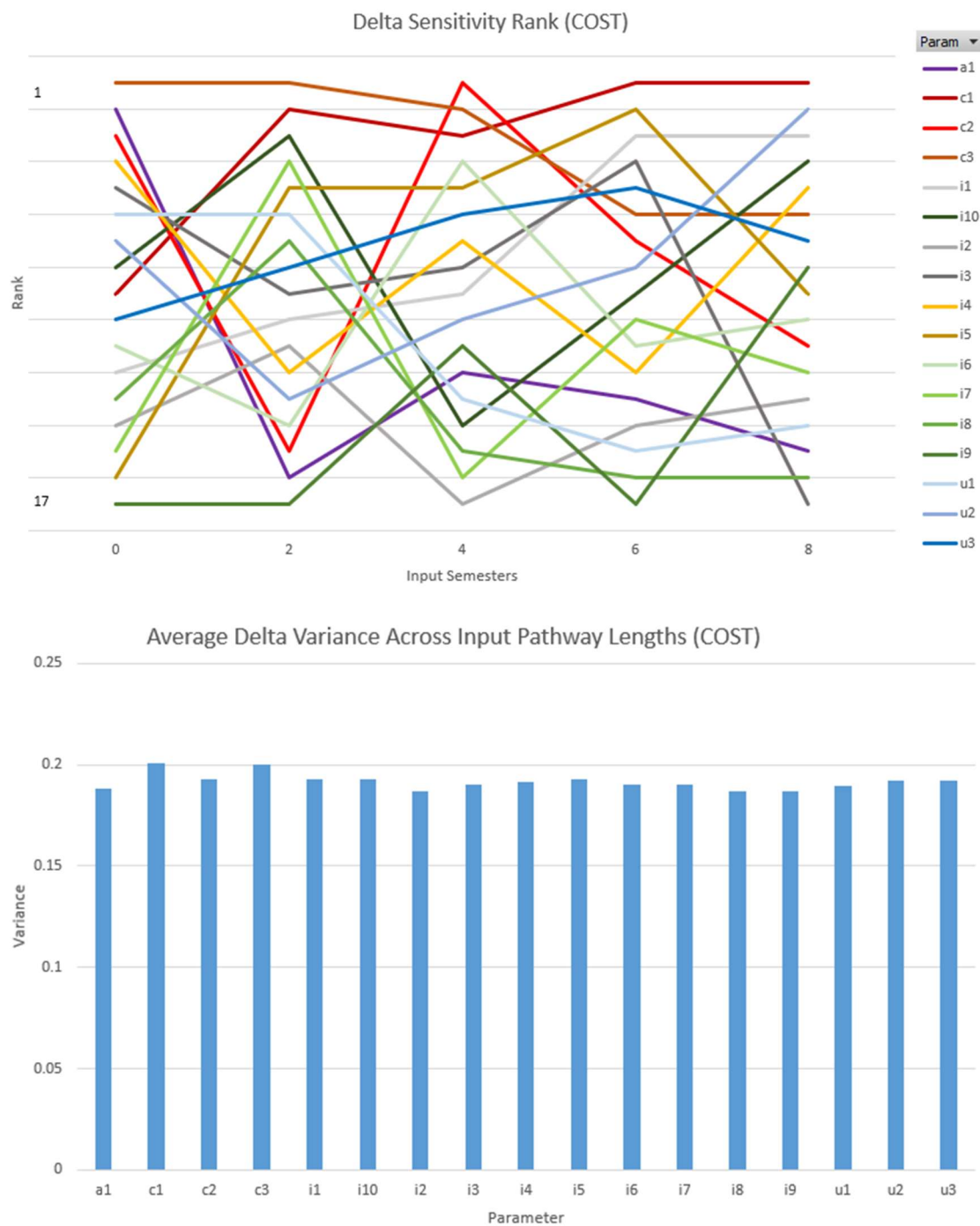


Figure 54. Delta analysis cost plots

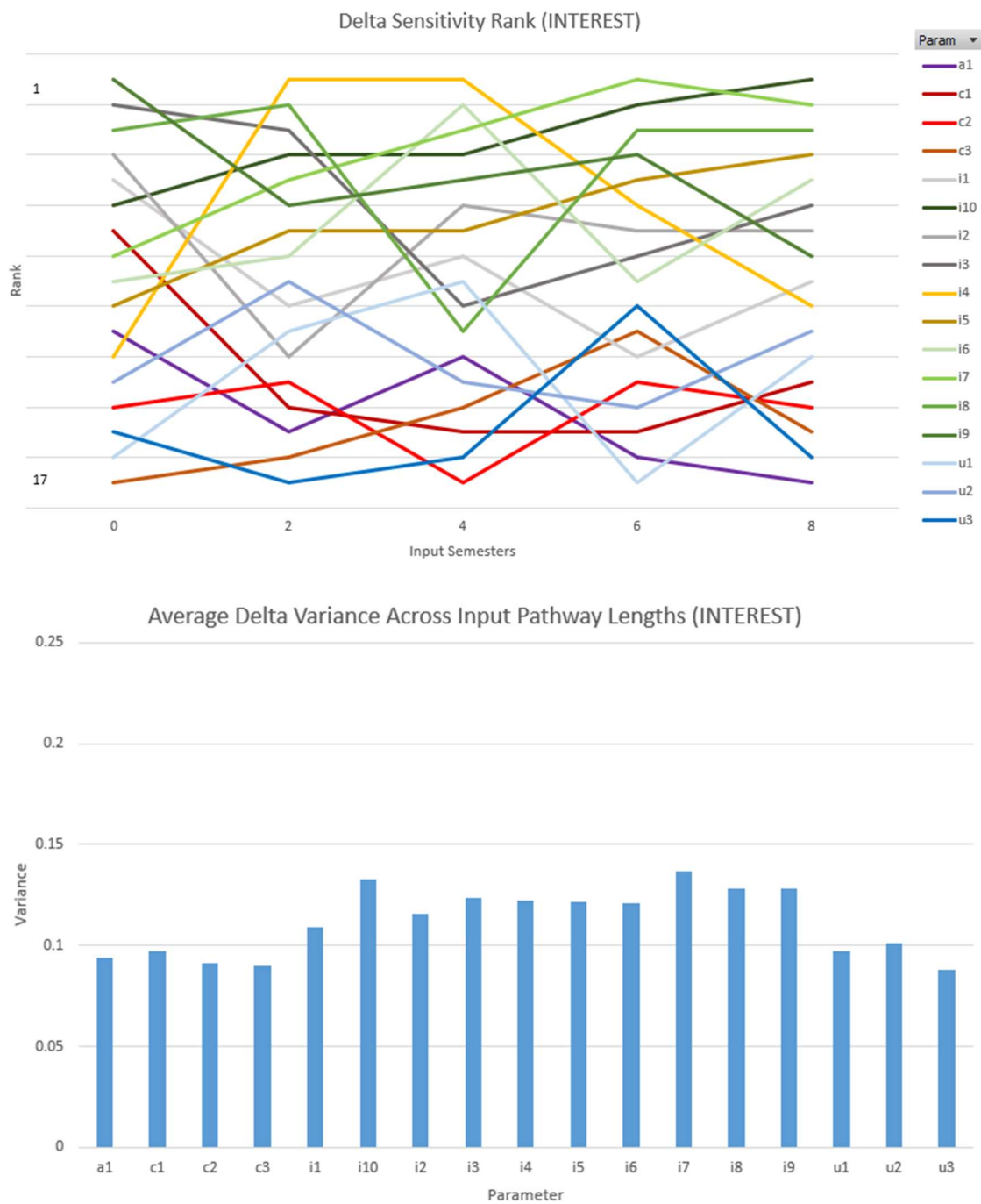


Figure 55. Delta analysis interest plots