January 2022

Improving Or Operations Using Machine Learning Techniques

Tannaz Khaleghi
Wayne State University

IMPROVING OR OPERATIONS USING MACHINE LEARNING TECHNIQUES

by

TANNAZ KHALEGHI

DISSERTATION

Submitted to the Graduate School

of Wayne State University,

Detroit, Michigan

in partial fulfilment of the requirements

for the degree of

DOCTOR OF PHILOSOPHY

2022

MAJOR: INDUSTRIAL & SYSTEMS ENGINEERING

Approved By:

____

Advisor Date

____

____

____

DEDICATION

I dedicate this dissertation to my precious daughter, Delara, my beloved spouse and my amazing mother for their endless love, encouragement and support. People who have always been there to support me, and show me the best path to follow.

ACKNOWLEDGEMENT

First and foremost, I would like to convey my heartfelt thanks to Dr. Alper Murat, my PhD adviser, for his unwavering encouragement, inspiration, and academic guidance during my studies. Aside from that, I would like to express my gratitude to the other members of the committee: Dr. Chinnam, Dr. Dalkiran, Dr. Arslanturk, and Dr. Kotov for their valuable input and assistance.

Finally, but certainly not least, I want to thank my family and friends for their unwavering love, encouragement, and support. I would not be who I am now if it were not for them, and this feat would not have been possible.

TABLE OF CONTENTS

Dedication . . . . **ii**

Acknowledgement . . . . **iii**

LIST OF FIGURES . . . . **vii**

LIST OF TABLES . . . . **ix**

1 Introduction **1**

1.1 Background 1

1.2 Statement of the Problem 2

1.2.1 Text Mining: Misspelling Correction and Abbreviation Detection in Healthcare 4

1.2.2 Prediction: Surgery CPT Code Prediction, single and multiple CPT(s) . 5

1.3 Surgery Duration Estimation 6

1.4 Research Questions and Objectives 7

1.5 Significance, Implications and Contributions 7

2 Literature Review **9**

2.1 Text Mining: Misspelling Correction and Abbreviation Detection in Healthcare . 9

2.2 Prediction: Surgery CPT Code Prediction and Duration Estimation 11

3 Automated Surgical Term Clustering: A text mining approach for surgery description unstructured data **15**

3.1 Preprocessing and cleaning 15

3.1.1 Removing special characters 15

3.1.2 Detecting stop words 16

3.1.3 Tokenization 16

3.1.4 Stemming & Stem Derivatives 17

3.1.5 Computing TFIDF Matrix 17

3.2 Calculating Similarity Matrix 18

3.2.1 Levenshtein Distance Matrix 18

3.2.2 Updated Levenshtein Distance Matrix 18

3.2.3 Optimal Distance Weight 20

3.3 Hierarchical Agglomerative Clustering (HAC) 23

3.4 Finding the best cutoff distance of the dendrogram 24

3.5 Heuristic Clustering of HAC (HCHAC) 26

3.6 Medical Text Analysis Results 29

3.6.1 Obtaining Ground Truth 29

3.6.2 Impact of Clustering Method 30

3.6.3 HAC and HCHAC Clustering Results 31

3.6.4 Choosing the best cutoff point 34

4 Classification: Primary and Multiple CPT Prediction 36

4.1 Primary CPT Prediction 36

4.2 Primary CPT Prediction with respect to Surgery Duration Loss 45

4.3 Multi-CPT Prediction 51

4.4 Evaluation Metric 56

4.5 CPT Prediction Results 58

4.5.1 CPT and Duration Prediction Results 58

4.5.2 Primary CPT prediction 58

4.5.3 Multi-CPT Prediction 62

5 Surgery Duration Prediction 63

5.1 Direct Method 63

5.2 Indirect Method 65

5.3 Evaluation Metrics 66

5.4 Duration Estimation Results 68

6 Conclusion **75**

6.1 Text Mining: Misspelling Correction and Abbreviation Detection in Healthcare . 75

6.2 Prediction: Surgery CPT Code Prediction, single and multiple CPT(s) 76

6.3 Surgery Duration Distribution Estimation 77

APPENDIX **78**

REFERENCES **85**

ABSTRACT **98**

LIST OF FIGURES

1 Figure 1. Algorithm represented for adding stem derivatives to the universal dictionary, SDOD. 17

2 Figure 2. Preprocessing Stages 18

3 Figure 3. Algorithm represented for generating the Levenshtein distance matrix [39]. 19

4 Figure 4. Flowchart represents the paths A, B, and C to update Levenshtein distance matrix. 19

5 Figure 5. Changing distance of abbreviations and complete form and its derivatives. 21

6 Figure 6. Removing the words while duplication of their full spelling word as an abbreviation exist in SDOD. 22

7 Figure 7. Generating sample of distances for cutting the dendrogram. 25

8 Figure 8. Selection sort algorithm as an initiative step for HCHAC. 26

9 Figure 9. HCHAC algorithm for empowering HAC. 27

11 Figure 10. HCHAC and HAC precision (worst and best scenarios) plot. 33

12 Figure 11. HCHAC and HAC recall (worst and best scenarios) plot. 33

13 Figure 12. HCHAC and HAC F1-score (worst and best scenarios) plot. 33

14 Figure 13. HCHAC and HAC accuracy (worst and best) scenarios plot. 33

15 Figure 14. Cluster Weights on different random dendrogram cut for each specialty. 34

16 Figure 15. P/R/F for different cuts per specialty. 35

17 Figure 16. Random Forest classification algorithm. 38

18 Figure 17. Extreme Gradient Boosting algorithm for multi-class prediction [15, 29]. 39

19 Figure 18. Algorithm for calculating class weight when word w is in both dictionaries, $CPT\_catalog\_dict_{p_i}$ and $CPT\_catalog\_dict_{actual}$. 42

20 Figure 19. Algorithm for calculating class weight when word w exists in $CPT\_catalog\_dict_{p_i}$ but not in $CPT\_catalog\_dict_{actual}$. 43

21 Figure 20. The class weight recalculation procedure. Algorithm 10 and 11 are calculating new weights based on the observed circumstances. 44

22 Figure 21. A simple GA structure. 48

23 Figure 22. GA crossover example. 49

24 Figure 23. GA mutation example. 49

25 Figure 24. Multi-channel CNN model architecture. 55

26 Figure 25. Most frequent CPTs distribution and average durations per specialty. . 58

27 Figure 26. Model comparison: plot of mean, median and standard deviation of actual durations for top 5 CPTs in Cardio specialty. 71

28 Figure 27. Model comparison: plot of mean, median and standard deviation of actual durations for top 5 CPTs in General specialty. 71

29 Figure 28. Model comparison: plot of mean, median and standard deviation of actual durations for top 5 CPTs in OBGYN specialty. 71

30 Figure 29. Model comparison: plot of mean, median and standard deviation of actual durations for top 5 CPTs in Urology specialty. 72

31 Figure 30. Model comparison: plot of mean, median and standard deviation of actual durations for top 5 CPTs in Other specialty. 72

32 Figure 31. Model comparison: plot of mean, median and standard deviation of actual durations for the CPT durations with wide standard deviation in OBGYN specialty. 73

33 Figure 32. Model comparison: plot of mean, median and standard deviation of actual durations for top 5 CPTs in Other specialty. 73

34 Figure 33. Model comparison: plot of mean, median and standard deviation of actual durations for top 5 CPTs in Other specialty. 73

35 Figure 34. Actual and scheduled duration plots for significant CPTs in specialty datasets. 80

36 Figure 35. Duration percentile of average predicted and actual durations in all specialties. 81

37 Figure 36. Kernel Density Estimate plot using Gaussian kernels to estimate and show the probability density function (PDF) of both actual and scheduled durations. 82

38 Figure 37. Heatmap of of average actual surgery duration per procedure start hour of surgeries (0-24) and CPT codes. 83

39 Figure 38. Heatmap of of average actual surgery duration specialty (pre-aggregation in terms of other) and CPT codes. 84

LIST OF TABLES

1 Table 1. Examples for paths A, B, and C. 22

1 Table 2. HCHAC complementary step for clusters grouping. In Table 2a you can see that 2 FPs and 4 TPs and in Table 2b 1 FP and 9 TPs are added to the results of HAC. Therefore, the gain of this method is 13 TPs as opposed to 3 FPs loss. . 28

2 Table 3. Study Framework and steps of this research along with methodologies and outputs. 28

3 Table 4. Insights about text data per specialty 29

4 Table 5. Feature reduction in each specialty with respect to the performance measure 32

5 Table 6. Confusion matrix in clustering 32

6 Table 7. The accuracy of abbreviation and typo detection in text 34

7 Table 8. Different settings (data filtering methods) for deep result analysis 45

8 Table 9. Multi-channel CNN model summary. 56

9 Table 10. CPT prediction accuracy measures under different filter combinations and accuracy calculation approaches for each specialty. 60

10 Table 11. Plot of average CPT prediction accuracy under optimal settings for XGB and RF models. 61

11 Table 12. Actual duration distribution estimation using different proposed alternatives. 66

12 Table 13. Duration estimation performance in different specialty datasets (point estimates vs duration distributions) for CPT known assumption and when the CPT is unknown and predicted in our presented models. 70

1 Introduction

1.1 Background

Predictive and prescriptive analytics are quickly forming a modern era in health care decision support system. Predictive models and machine learning methods are well-developed with a considerable impact on many industries including healthcare delivery systems. Medical service providers are increasingly benefiting from the remarkable developments in this domain by improving quality of care, hospital administrative decisions, disease management and predictions, and health care supply chain efficiency. One of the under-investigated opportunities in health care studies is the study of the predictive analytics applications and their integration with well-studied prescriptive methods for surgical services.

Prescriptive analytics aim at providing (near-)optimal solutions for decision problems using optimization techniques. These methods enable healthcare planners and analysts to recommend the best course of action for care providers and patients. In the context of apriori planning of healthcare operations, these methods provide solutions to operational problems by using deterministic and stochastic approach. In addition they can be used as a comprehensive tool for comparing multiple scenarios to foresee the effect of selecting one decision over another through “what if" analysis. However, in this research, we explored another benefit of such methods. It’s inevitable that optimization models help in selection of the optimal parameters for machine learning tasks. Given the power of machine learning techniques in learning patterns and providing most accurate predictions, their case-by-case weaknesses are causes of higher prediction errors. To alleviate this, prescriptive models further improve the outcomes by filling the gaps within each ML model. Today, predictive and prescriptive technologies together offer a more evidence-based and transparent approach to decision making.

In recent years, the growing rate of inefficiencies caused by scheduling problems in surgery departments highlights the need for more potent decision support tools. In surgical services, misalignment between what is planned for and when versus what actually happens and when creates stressful atmosphere both for staffs and patients. In this research, we develop an analytical framework as a solution to this problem using both prediction and prescriptive approaches. Surgical procedure durations are innately unpredictable attributable to known and unknown patient, surgical team and other factors. Further, the degree of uncertainty varies across different surgery categories. Effective planning of surgical service operations thus necessitates accurate prediction of surgery demand (duration, equipment, etc. [45]) and is a key step in efficient scheduling of operating rooms.

Developing a robust system for accurately predicting surgery durations is challenging and requires use of different learning methods. Various predictive learning concepts aligned with health care system goals can bring outcomes that not only satisfy the staffs' expectations but also improve the patient experience in surgical unit, if presented and utilized accordingly. Generally, the main focus of related research is to improve surgical service processes and make them more predictable [45]. In addition to other predictive approaches, text mining methods offer significant improvement in predicting and planning surgical service operations in hospitals. [46].

A surgery is characterized by a set of vital activities that is characterized by single or multiple codes from a list of surgical procedure codes (Current Procedure Terminologies, CPTs) maintained by the American Medical Association for singularity and consistency. CPT codes are known to be one of the most significant determinant in estimating the duration of the surgeries [85]. Due to the various uncertainties, exact characterization of surgical activities for a given case is unknown apriori. Instead, various text-based perioperative notes (entered by providers, nurses and schedulers) as well as other information available through medical records specific to the patient, condition and providers are available. Hence, ability to use multiple types of input information is critical in predicting the CPT codes and surgery durations for effective operatoins planning. In this context, integrating text mining approaches to extract improved feature set in predicting CPT codes of surgery cases and surgery durations is critical.

1.2 Statement of the Problem

Over the past decade, there has been an increased effort in both the academia and practice to develop optimized solutions for the operating room scheduling problem. OR, as an expensive facility in hospitals, requires a continuous performance control system to ensure high utilization of expensive resources and while minimizing delays due to different sources. A major source of such delays is the misalignment between planned and realized operations. One major contributor to such misalignment is relying solely on prescriptive analytics, e.g. deterministic optimization methods, that fails to account for the implications of variations of daily surgical cases on the planning decisions. While there are more advanced prescriptive approaches that recognize the variability, i.e., stochastic programming, their effectiveness depends heavily on how well the surgery duration variability is captured. Hence, with this dissertation, our main goal is to improve the operational planning decision making in surgical services with novel predictive approaches that can more accurately characterize the variability of surgery durations. In this research, we use and develop novel supervised and unsupervised methods to improve the CPT and duration prediction task. In this problem, given a set of preoperative features including unstructured surgery descriptions, we first perform feature engineering and create features for the machine learning model from raw description data. Then, we build a robust framework for predicting CPT code and surgery durations using both predictive and prescriptive models. The outcome of this research helps extensively in scheduling and billing processes.

This research consists of three parts: unstructured text mining to generate useful features for prediction task, classification approaches for surgical operations planning and estimating / predicting surgery case duration based on CPT classification outcome. In the first part, this research investigates the prediction of the surgery durations and Current Procedural Terminology (CPT) Codes. Accurate prediction of the surgery duration improves the utilization of indispensable surgical resources such as surgeons, nurses, and operating rooms. Prediction of the correct CPT codes not only helps the preparation process for the surgery (i.e., case cart) but also enhances prediction of surgery duration distributions. State-of-the-art efforts, albeit without text features, have predicted the point estimates of surgery durations given the set of predictive features. As an alternative to point-estimate (i.e., direct approach), we also evaluate the surgery duration prediction performance through the predicted CPTs in a two-step approach where the surgery duration distributions are estimated from the predicted CPT codes. In another part of the dissertation, we improve the single PCT prediction task by optimizing tree selection with respect to two goals; CPT classification and duration prediction. This enables the user to make hybrid decision while both targets are counted in the final decision to a specific level of importance. Furthermore, we predict secondary CPT using multi-channel deep neural network structure using text, categorical, and continuous data. By integrating duration estimation (predictive) and case schedule optimization (prescriptive) tasks, we develop a hybrid method to improve resource utilization by more accurate prediction of important indicators in efficient surgery planning. This research contributions are further discussed in the following subsections.

1.2.1 Text Mining: Misspelling Correction and Abbreviation Detection in Healthcare

Text data set's ingestion for statistical predictive modeling requires preprocessing and transformation steps. Hospital staff usually enter the surgery procedure descriptions and other information without a predefined pattern, i.e., mostly manual entry. Hence, a text input referring to same surgery procedure with the same current procedural terminology (CPT) code might be entered differently (different order of terms, abbreviations, typos). Therefore, when estimating durations considering text as one of the features may mislead the prediction instead of providing value. In medical domain, correcting misspelled words is a crucial task to ensure reliable interpretability of medical records. Additionally, natural language processing practices in information retrieval such as knowledge extraction and information encoding are preconditioned on existence of a solid tool to appropriately detect and correct the possible data noises.

In this research, the initial step is to organize the unstructured text data with text mining approaches to reduce the unnecessary variability of the descriptions and increase the efficiency of our predictive model. It is also worth mentioning that, the provided procedure descriptions precisely characterize the surgery case process, but it can bring some pitfalls in regards of surgery procedure code due to additional minor details for each case. Particularly, in health-related research, text data is mostly being used where the symptoms of the diseases should be extracted to find the best cure for them using text mining approaches. Recently there has been some developments regarding application of text mining in hospital's claim processing and DNA/gene pattern recognition. In our approach a mixture of measurements and concepts originally generated from text mining methods leads us to distinguishing transformed features.

The corpus in our dataset contains 17,400 rows of surgery cases, and consequently, the same amount of textual procedure descriptions each of which describes a case scheduled for a patient in a particular day of surgery at operating rooms (ORs). Text mining approaches has been deployed to help us achieve our goal and gradually improve the subgroups of medical terms. Following several steps of cleaning and organizing the text, we try to form the initial clusters of the medical terms using a weighted Levenshtein distance matrix and Hierarchical Agglomerative Clustering (HAC) method. The second phase of clustering is developed and conducted using a heuristic clustering approach by N-grams distance which we refer to as the Heuristic Clustering of Hierarchical Agglomerative clusters or (HCHAC) which is triggering the reduction of true negative cluster members in our first attempt of clustering results. This method is designed specifically to the group the abbreviated terms and its complete format in the same cluster.

Once our text analysis is accomplished, it provides the chance of reviewing the outcome of clustering the typos, abbreviations, and raw medical terms based on a customized sorting technique. The primary benefits of our approach is that there is no need for identifying the number of clusters in the first phase of clustering and the distance between the clusters provides rational measurement which is setting the differences between the terms reasonably. The distance matrix is a norm of modified Levenshtein distance to account for the abbreviation forms of the words more precisely. We demonstrate effectiveness of our approach empirically, not only for clustering medical text but also for identifying underlying structure of clusters, using real world datasets.

1.2.2 Prediction: Surgery CPT Code Prediction, single and multiple CPT(s)

While some surgical services use a preliminary CPT code(s) prior to the surgery for operational planning, majority of services choose to only use only textual descriptions and identifiers. Even those providers using preliminary CPT codes end up revising their CPT codes upon the completion of the surgery by examining the surgery notes (i.e., medical coding for billing). Hence, apriori prediction of CPT code(s) for a surgery is critical for operations planning such as scheduling and equipment provisioning. Accurately predicting CPT code(s) not only help surgical services to prepare the necessary case equipment and other service resources, but also help the case scheduling.

Hence, this research contribution consists of CPT code(s) prediction and surgery duration estimation. For the CPT code(s) prediction, we evaluate the value of the text features (with and without dimensionality reduction as proposed in the previous subsection). Initially, our focus is on the primary CPT prediction and evaluate the predictive performances of different filtering and set-based prediction strategies. Tree-based classification models are proposed and enhanced to predict the primary CPT. While the objective function in a classification task is strictly defined as producing the correct categorical label, the importance of duration error is ignored. We, further, improve the framework by optimizing the decision tree selection in the boosting model to account for duration loss. We also let users decide on the outcome based on the classification and duration estimation goals by assigning weights to both targets.

While the primary CPT code is the most important determinant of surgery durations and preoperative planning tasks, surgeries often entail multiple procedures (i.e., auxiliary CPTs) which influence the surgery durations. Hence, by using multi-task learning concepts in the context of deep learning models, we aim to predict multiple CPT codes, i.e. set containing the CPTs of all procedures being performed in the operation. The predicted CPT set is then used to enhance the surgery duration estimation task. An accurate CPT set prediction may help scheduler to schedule OR in a more efficient manner.

1.3 Surgery Duration Estimation

For the surgery duration prediction, we first use all the available information (quantitative, categorical, and text features) and directly predict the surgery duration as a point-estimate using different regression models and develop estimates of variability using sampling methods to obtain distributions of the surgery durations (referred as "direct approach"). Next, we follow a two-step approach, where we first predict CPT code (either single or multiple based on previous step) as a multi-class classification task and then using the predicted CPT code(s), we develop empirical duration distributions. The direct and two-step approaches are compared for their bias in the mean duration estimation and ability to accurately characterize the duration distribution. Ultimately, the duration distributions are generated from both primary CPT prediction and CPT set prediction. It also worths to mention that the enhanced hybrid CPT classification method is also another alternative to generate duration distribution and use them for comparison purposes.

1.4 Research Questions and Objectives

This research's overall goal is to improve the prescriptive tasks of operational planning in surgical services with novel predictive approaches that can more accurately characterize the variability of surgery durations. Specific research questions are as below:

1. How to reduce the dimensionality of text features in surgical services through a dictionary-free correction of varied typo and abbreviations to improve subsequent prediction tasks?
2. How accurately and robustly the CPT code(s) can be predicted with and without text features?
   (a) What is the contribution of text features on CPT code(s) prediction?
   (b) How to best incorporate duration accuracy in the prediction of CPT code(s) using cost-sensitive learning concepts?
   (c) What is the best multi-task predictive strategy for the primary and auxiliary CPT codes of a given surgery?
3. How to best characterize the duration distribution of surgical cases apriori?
   (a) What are the advantages/disadvantages of point-based prediction versus two-step approach of first predicting CPTs and then characterize the duration distributions (in terms of prediction bias, practical utility, distribution characterization)?
   (b) How does prediction of multiple-CPTs versus only the primary CPT influence the duration prediction task?
4. How does the primary CPT prediction performs in terms of primary CPT classification accuracy and surgery case duration error?

1.5 Significance, Implications and Contributions

Significance and contributions of this research is as follows: Objective 1 of this research is relevant to those applications with prediction task entailing text features and the availability and use of a reference dictionary for abbreviations and typos is limited. While there exists medical and non-medical domain focused text mining methods for handling typo and abbreviations, all of them require availability of a reference (e.g., dictionary of common typos). Thus our contribution is the first dictionary-free approach for such tasks in the healthcare domain.

Objective 2 contributes to primarily healthcare application domain where a patient service (i.e. surgery) is defined in terms of single or multiple classes (i.e. CPT codes) and collectively determine a dependent attribute (i.e., surgery duration) of which either the mean value or the distribution (mean, median, standard deviation) is of concern. This approach contributes to the healthcare application domain as well as to the broad machine learning literature of cost-sensitive primary-class classification where the objective function of classification problem is combination of duration and classification loss function with weight assignment for optimal hybrid decision making. Also in the context of classification model another important contribution highlights the robustness of the optimized tree based machine learning model using Genetic Algorithm based on hybrid decision factors.

Objective 3 is relevant for healthcare and other domain applications where a continuous prediction task with a need for characterizing variability around mean (or median) prediction is decomposed into first a classification task and then subsequent step of distribution fitting based on the CPT class. This objective contributes to the healthcare predictive analytics domain as well as broad machine learning domain in terms of regression predictions and characterization of variability around mean estimates in the presence of distributional relationship with class membership. There are many healthcare and other manufacturing and service operations that rely on prediction of inputs fed into a subsequent prescriptive task for planning and execution.

In the remainder, we investigate some of the comprehensive studies in the application of text mining and prediction in health care efficiency improvement, our findings of applying our prediction methods to the surgical dataset in this dissertation research, and the future opportunities and developments in this area. The rest of this dissertation is organized as follows. In the next section, we review the related efforts to unstructured text analysis, and surgery CPT prediction and duration estimation. This section is followed by methodology section, in which the details about the research for aforementioned objectives are explained. Next section presents the results of the analysis framework, the dataset of our experiments and derived insights and lastly we conclude the research.

We fine tune the GA for crossover, mutation, fitness calculation and obtain the optimal results in terms of duration estimation.

2 Literature Review

This study adopts a systematic approach to investigate the literature related to surgery improvement potentials and problems which can be attributed to a specific class of surgery duration prediction and scheduling problem. We divide this section into subsections to review the related work of each step of the research.

2.1 Text Mining: Misspelling Correction and Abbreviation Detection in Healthcare

Many research studies have focused on the extraction of context information in healthcare systems as it is widely being generated in such system's databases. Moreover, the shortness of standardized medical text coding is a major difficulty that blocks improvements in the process of further automated learning tasks in subdivisions of any healthcare system. A short overview of such researches that investigate the importance of textual data and its alignments to the modern analytic settings is provided in this subsection.

In order to highlight key health related information from unstructured or semi-structured text data, text mining approaches has been used for over 35 years on different health records to extract medication, cancer, procedure, or other patient-related information [28, 72, 82]. These studies rely on many semantic, syntactic or lexical methods to identify and link different concepts [99]. The outcome of such studies reveals the potential of text mining to enhance data collection, improve quality of care, decrease the costs and possible risks arise from human errors [16, 93].

A remarkable group of researchers make extensive use of the existing standard vocabularies gathered by the National Library of Medicine in the unified medical language system (UMLS) while extracting information from their medical datasets. In 2008, a study was conducted by a group of researchers at Alabama University focusing on text mining tools for extracting critical information from textual data context. During their analysis they recognized that existence of some common words such as glass, pm and volume skewed their output clusters by unavoidable noise words that were tectonically false positive/false negative cluster members. An attempt for reducing this noise was beginning with a dedicated start list which helps distinguishing between technical and common terms in a clinical text. They used the standard vocabularies in UMLS to target some technical terms. [75]

Several works have investigated the issue of typo correction thus far, however, literature on misspelling detection/correction in medical notes as an explicit problem is indeed sparse [73]. A previous study generated a prototype spell checker by the use of Wordnet and UMLS as their sources of information [89]. Mykowiecka and Marciniak also designed a framework for automatic spelling correction in mammography reports using edit distance measure and bi-gram probabilities [65]. However, these methods are applied to a very specific domain. Kenneth H. Lai et. al. developed a spelling correction system for noisy medical text to make them correctly interpretable based on UMLS vocabulary source. The base model used in their system is Shannon's noisy channel along with vocabularies from different sources such as SPECIALIST lexicon (UMLS edition 2014), RxNorm Drug lexicon (April 2014), list of previously observed abbreviations, and Aspell's English dictionary. Such libraries need continuous update in terms of new words and their different forms of typos, misspellings and limited amount of synonyms while the methods offered in this study helps eliminating the dependency of most research questions in this area to an upgradeable library to a great deal. [52]

Other than the dictionary-based methods, some studies devised unsupervised text mining methods such as affinity propagation or its hierarchical extension, and the family of K-means (e.g. K-medoid and K-median)[27, 32, 38] and applied them to medical text data to identify the similar terms (both contextually and syntactically), analyze HIV-strain mutation and detect genes. Nonetheless, most of these approaches necessitates some restrictions on the data and initialization steps, therefore, they are not recommended for an automated framework of clustering. Moreover, these methods identify data point differences based on similarity measures such as N-grams, edit distance, cosine and so on. Anna H. compares and analyzes the effectiveness of some distance measures in such algorithm for some selective text documents. The results of this study shows the outcome on seven text document datasets and five most popular distance methods in text clustering. [40]

In 2017, Kruse, Eiken and Vestergaard studied establishing patient groups of high, average and low fracture risk by an unsupervised machine learning algorithm. They studied standardized variable means, Euclidean distances and Ward's D2 method of hierarchical agglomerative clustering (HAC), on regional and national Danish patient data on dual-energy X-ray absorptiometry (DXA) scans and medication reimbursement to form the clusters based on bone mineral density characteristics. Using this method nine clusters were obtained which represents this approach as a novel tool for enhancing patient characteristics in bone disease beyond traditional diagnosis and current DXA scan indication guidelines can be further improved by HAC algorithm. [51] Their study shows the effectiveness of clustering function of HAC algorithm in medical contexts. However, present study improved the first stage of clustering method by applying HAC with a weighted Levenshtein distance and by developing a second phase of clustering method which tends to combine the initially cluster outputs that are not merged by HAC but both are representing the same term.

2.2 Prediction: Surgery CPT Code Prediction and Duration Estimation

In health care, patient safety matters the most in every aspect of the patient related operations in this system. The reliable safety level needs to be ensured by the staff at any level of authority. Machine learning methods are well-known tools in knowledge diffusion from raw data in many areas including health centers. Such methods solve many physicians' challenges in different departments such as accessing to up-to-date clinical evidence, patient-related information interaction and details of personalized medicine. More specifically, supervised learning methods can support human controlled prediction studies such as case identification code assignment, operation duration forecasting, and risk assessment. The duration prediction studies are widely being used and they are not limited to operating rooms in healthcare. [94] evaluate the performance in an automated application for classification of mesothelioma patient's personal and family history of cancer from clinical reports.

Assigning the International Classification of Diseases in patient visit data (ICD code classification) is a similar challenge to CPT code classification in terms of the input types and the problem structure. [56] present a Multi-Filter Residual Convolutional NN in ICD classification task. Their main contribution includes a multi-filter convolutional layer to learn different patterns in textual data with various lengths and residual convolutional layer to expand the receptive field. [96] developed ML models that can handle unstructured, semi-structured and structured text data from different modalities. The final ICD code assignments are predicted through the ensemble method of ML models. In another study, [30] propose a vector-space based topic modeling to prepare clinical data by using fuzzy similarity-based cleansing method to capture redundant patient data. Moreover, various multi-label classification models are used to facilitate ICD coding task [9]. [78] offer a hierarchical deep learning classifier that uses CNN model to employ in the problem of pathology reports with applicable 9 unique ICD morphology codes. They demonstrate that the hierarchical deep learning classification method improves on performance in comparison to a flat multi-class Convolutional Neural Network model for ICD morphology classification problem.

Many studies classified the medical radiology reports as textual data using deep neural network architectures. [8] compare two deep NN structures, an attention-based hierarchical recurrent neural network (RNN) and a CNN Word Glove to the domain specific rule based system (PEFinder) and to other machine learning models such as SVM and Adaboost in radiology report classification problem. The results suggest that the deep NN outperforms traditional methods given the single institutional training dataset. [55] represent RNN system for automatic prediction of binary class; fracture and non-fracture cases. Their results show that the proposed RNN system classifies important findings in radiology reports with %96 F1-score. [88] developed a multi label classification system, which includes a text feature engineering, feature reduction and multiple classifiers, for automatic diagnostic code assignment to radiology reports. Accordingly, there are quality researches to establish valuable domain-specific embedding sources and provide useful pre-trained biomedical word embeddings which has extensively been used in deep NN based papers [17, 42, 77].

Many researches try to predict accurate durations in emergency rooms, and clinics to reduce the patient wait times in vital departments. In 2009, Stepaniak et. al investigates the possibility of existence of correlation between some surgeon related factors (such as age, experience, gender, and team composition) and procedure durations [83].

In the presented literature, several studies have focused on predicting the remaining surgery duration (or RSD) intra-operatively. The main challenge of such researches is predicting surgery durations prior to the scheduled time of operation using some correlated preoperative factors such as surgeon, procedure descriptions, case type, and other intra/inter-operation circumstances. These features combined with proper prediction model with consideration of preoperative variations can provide valuable information regarding the operation durations. For instance, Travis et al. have demonstrated the consequence of underestimated anesthesiology durations on the total wait times [90].

Also, a great variation in waiting durations with mean of 47 and deviation of 17 minutes is reported for 157 patients of cholecystectomy case by Paalvast et al., while it has been shown that the patient preparation time is usually around 25 minutes. They also proposed a classification method using the activation of the electro-surgical devices to arrange the upcoming patients reception [70]. In 2009, Dexter's work focused a semi-automatic method which requires the input from anesthesiologists during the surgery [20]. Other indications, such as surgical tool utilization [62, 71], and low-level operation representations [26] have also been used to estimate RSD of surgery rooms.

In the computer vision community, only a few studies address the problem of estimating the RSD. For instance, a recent attempt proposed an architecture to localize short length activities (e.g. tennis swing, and cliff diving). Their method helped in predicting such activities completion progress as well [10]. In Li et al. study, a deep architecture was deployed to do progress approximation and phase identification from various datasets, and as a result the remaining duration is subsequently derived from the progress [57].

From another view point, predicting reliable elective surgical cases durations is quite a challenging task which is available in most of the studied extensively [79]. Many authors have pointed three main trends out in this area. The first group of researches, explore the significance of some elements, such as procedure interruptions, communication failures, and team familiarity that contribute to the duration variations in many processes [14, 31]. Studies of the second stream investigate the goodness of fit of the known distributions for estimating procedure durations, especially the normal distribution and lognormal distribution [81, 83, 86].

The next trend of papers that tend to develop predictive models for surgery duration using machine learning models such as regression methods with continuous prediction labels. For instance, Combes used rough sets and neural networks to predict the durations [18]. Along with these point estimate approaches, Dexter (2013) also found upper prediction limits for surgery durations [21]. They find that factors such as team composition, experience, and time of the day are significant elements in estimating surgery duration variable [22, 44, 58, 84]. The results of the studies in this stream are derived from a particular specialty. For instance, in 2001 Pisano et al. investigated the rate of improvement from cumulative experience for cardiac surgeries [74]. Later on, in 2005 Ballantyne et al. analyzed the learning curve for gastric bypass operations [7].

With respect to the revised cost-functions in machine learning models (i.e. tree based models) to control over predominant categorical and continuous decision factors, [67] take a comprehensive investigation for choice of representative cost functions and analyse their latent properties. [95] derive a semantic loss function which defines sensible connection between neural output vectors and the model's logical constraints. Their proposed loss function represents how close NN is to fulfilling the constraints on its results. [37] demonstrate robust regularized extreme ML frameworks to reduce the effect of noise by using the asymmetric and Huber loss functions. The literature is limited in using custom loss function in tree-based models due to the challenge in revising the Gini index measure of the tree splits based on the features. In this paper, we support the bagging step in the Random Forest algorithm and optimize tree subset selection using GA given the duration loss function.

The last stream includes papers which used optimization methods in machine learning models, [69] represents an open-source GA based AutoML package that optimizes feature selections in preprocessing step and machine learning models with the goal of maximizing classification accuracy. More specifically, in the context of healthcare studies, [48] performed a classification task using different classification algorithms optimized by Particle Swarm Optimization (PSO) combined with Ant Colony Optimization (ACO) approaches. [59, 61, 66, 87, 97] optimized the input parameters of the Random Forest model based on optimization methods e.g. Genetic and Bayes algorithms, and etc.

In a wide range of real-world applications, many researchers improved the optimization experience by incorporating machine learning methods to predict the stochastic optimization solutions. Nowak and Emami et al. [23, 68] approximated a double stochastic matrix in the output directly derived from the neural network to characterize the permutation. In another study, Larsen et al. [53] predicted the solution to a stochastic load planning problem with a given deterministic mixed-integer linear program formulation by training a neural network model. They proposed a simple feed-forward NN structure for processing the input vector, and incorporating limited prior knowledge about the structure of the stochastic problem.

3 Automated Surgical Term Clustering: A text mining approach for surgery description unstructured data

As described in the "introduction" chapter, this research is divided into individual yet unified study steps. In the first section, we define and explain the text mining methodology with precise details in dedicated subsections. Also, we discuss how the outcome of text mining analysis is used in the next step; CPT prediction task. We have brought common initiative ideas of clustering and combined them with some concepts to form a unified system for conducting more robust and accurate clustering analysis. To explain this solution system, we divide the integrated methods into the following subsections and will explain each in detail.

3.1 Preprocessing and cleaning

Data collection approaches are sometimes loosely controlled which associates with missing values, wrong entries and outliers due to system malfunction or human error. Therefore, in data mining and knowledge discovery case studies the initial step is to perform data preprocessing on raw data. It is crucial to determine the framework of the analysis input and prepare high quality data prior to any kind of mining process. Likewise, we will precisely define the phases of the cleaning procedure and how it improves the clustering results. Many researchers [5, 91] have studied the influence of preprocessing stages specifically in the area of text mining. In addition to authors who claimed that feature selection [25] and feature extraction [35] affect the classification or clustering process significantly, we also underline the importance of preprocessing phase that not only can change the results of text clustering positively but it also helps improving further prediction phase results by pruning unnecessary features. The preprocessing stage usually can be categorized into functions such as filtering, lemmatization, tokenization, stemming and data transformation. In the next section, table 3 shows the effect of preprocessing steps on removing less important words.

3.1.1 Removing special characters

We coded the process of removing unwanted characters, numbers and words by using natural language processing techniques and regular expression operations. The functions are able to handle some frequently used technical abbreviations in the surgery descriptions such as I&D and D&C which stand for "Incision and Drainage" and "Dilatation and Curettage" respectively.

Also as the roles of employees in any organization may change continually and without a predefined pattern these descriptions end up containing high level of noise and outliers. As a consequence the more variation is added to the text, the more complex preprocessing steps need to be designed. The special characters such as:

1. Slash, backslash, dash, semicolon and so on are replaced with space since the later steps can handle any number of spaces between the words.
2. Dot is not removed but is retained to determine the abbreviation form of words. These abbreviations form "Abbreviation Dictionary" (AD). In some cases, two words are attached to each other by a dot or dot comes as the first letter of a word. By defining different regular expression compilers we are able to handle such errors that are associated with dot sign more efficiently.
3. The words with length less than three excluding some meaningful symbols such as "LT" or "L", "RT" or "R", and "FT" are removed from the text.

3.1.2 Detecting stop words

Stop words are terms that need to be eliminated prior to processing of textual data. Although one might think that the words appearing more frequently in a text are more likely to be considered as important terms but eventually such words are less informative and usually refers to the common words in a language. Since available tools fail in removing all stop words to support phrase search expertly, such extra efforts are essential for accomplishing a ready-to-use text. The stop word list used in this study contains around 130 stop words of English language and added 40 frequently observed medical stop words such as dr, surgery, doctor, treatment, and so on.

3.1.3 Tokenization

In tokenization stage, textual content is breakdown into words, symbols, terms, and etc. The output of this stage is then a list of words which can be used as input for further processing of text mining. After tokenizing the descriptions into words in the text, the abbreviated form of words and typos with their corresponding frequencies are stored in a dictionary, called "Original Term Dictionary" (OD).

Algorithm 1 Add-stem-derivatives(SDOD)

```
procedure Stem-Derivate($w$)
    for each word $w$ in SDOD do
        $L1 = Length(w)$
        $w' = Stem(w)$
        $L2 = Length(w')$
        $T = 1000$
        for each character $ch$ in $w \setminus w'$ do
            $w'_D = w' + ch$
            Add $w'_D$ to SDOD
            SDOD$[w'_D] = T$
            $T+ = 1$
        end for
    end for
    Return SDOD
end procedure
```

Figure 1. Algorithm represented for adding stem derivatives to the universal dictionary, SDOD.

3.1.4 Stemming & Stem Derivatives

Finding the roots of the existing terms in a dictionary helps reducing the dictionary size as two or more words may share the same root. However, this method cannot treat the words with typos or abbreviated words in the same way. Therefore, we added the stemmed version of the words and the derivatives of them using stem-derive algorithm represented in Figure 1. We will explain about the benefits it offers to our studies in the next step. Furthermore, the processed words obtained by each step will be added to a dictionary which we call "Universal Dictionary" (SDOD) for further analysis. Note that the counts of the stem derivatives is greater than 1000. It enables us to map the derivatives and original words in two dictionaries, Universal Dictionary (SDOD) and Original Term Dictionary (OD).

3.1.5 Computing TFIDF Matrix

A numerical statistic that reflects the importance level of a word in each document of corpus is called term frequency - inverse document frequency (TF-IDF) described by LP Jing [43] in 2002. TFIDF can be categorized in feature selection methods and is widely applied in text mining studies with the purpose of increasing the system's efficiency and avoid over-fitting [100]. This method consists of two parts, first Term Frequency (TF) as the frequency of occurrence of a term in a document and second Inverse Document Frequency (IDF) as a penalty weight of terms based on their frequency of occurring in the corpus. After constructing the clusters of the words, we can use this method to get the importance score of the words in each surgery description. The TFIDF matrix provide fruitful input information for predicting CPT Code of the surgeries in a supervised manner.

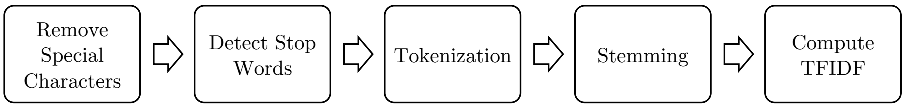


Figure 2. Preprocessing Stages

3.2 Calculating Similarity Matrix

3.2.1 Levenshtein Distance Matrix

The Levenshtein distance, also called edit distance introduced in 1965, is a sensitive measure by which between-string distances (in this case hand typed medical terms) are calculated. This algorithm tends to assess the cost of the least expensive set of operations such as insertions, substitutions and deletions for transforming one word into another [39]. The recursive Levenshtein algorithm that takes two strings, s_1 and s_2, and returns the corresponding distance between them is provided from Wilbert's work [39] as shown in Figure 3. Among the sting-based distance methods such as cosine, Jaccard, euclidean similarity measures [34] this method is more effective in terms of spell checking when we have many typos due to manual entries by passing in various types of operations.

3.2.2 Updated Levenshtein Distance Matrix

Levenshtein distance matrix does not account for abbreviated forms of the terms as it dynamically searches for minimum amount of change can be made to transform a term into another. Consequently, it can report high dissimilarity between such pairs (in some cases distance of 1) which is undesirable, hence the goal of clustering terms with consideration of typos, abbreviations, and stem derivatives will be violated. To tackle this issue, we define possible scenarios for the terms in original dictionary (OD) and target the expected responses per scenario by the represented paths in Figure 4.

Algorithm 2 Levenshtein-Distance(S1,S2)

```
procedure LEVENSHTEIN-DISTANCE($S$)
    for $i = 0$ to $m$ do
        for $j = 0$ to $n$ do
            $Upper = UpperLeft = left = maxint$
            if $i > 0$ then $Upper = dist[i-1, j] + weight(S1[i], \emptyset)$
            end if
            if $i > 0$ and $j > 0$ then $UpperLeft = dist[i-1, j-1] + weight(S1[i], S2[j])$
            end if
            if $j > 0$ then $left = dist[i, j-1] + weight(\emptyset, S2[j])$
            end if
            $dist[i, j] = min(upper, upperleft, left)$
            if $dist[i, j] = maxint$ then $dist[i, j] = 0$
            end if
        end for
    end for
    Levenshtein-Distance=$dist[m, n]$
    Return Levenshtein-Distance
end procedure
```

Figure 3. Algorithm represented for generating the Levenshtein distance matrix [39].

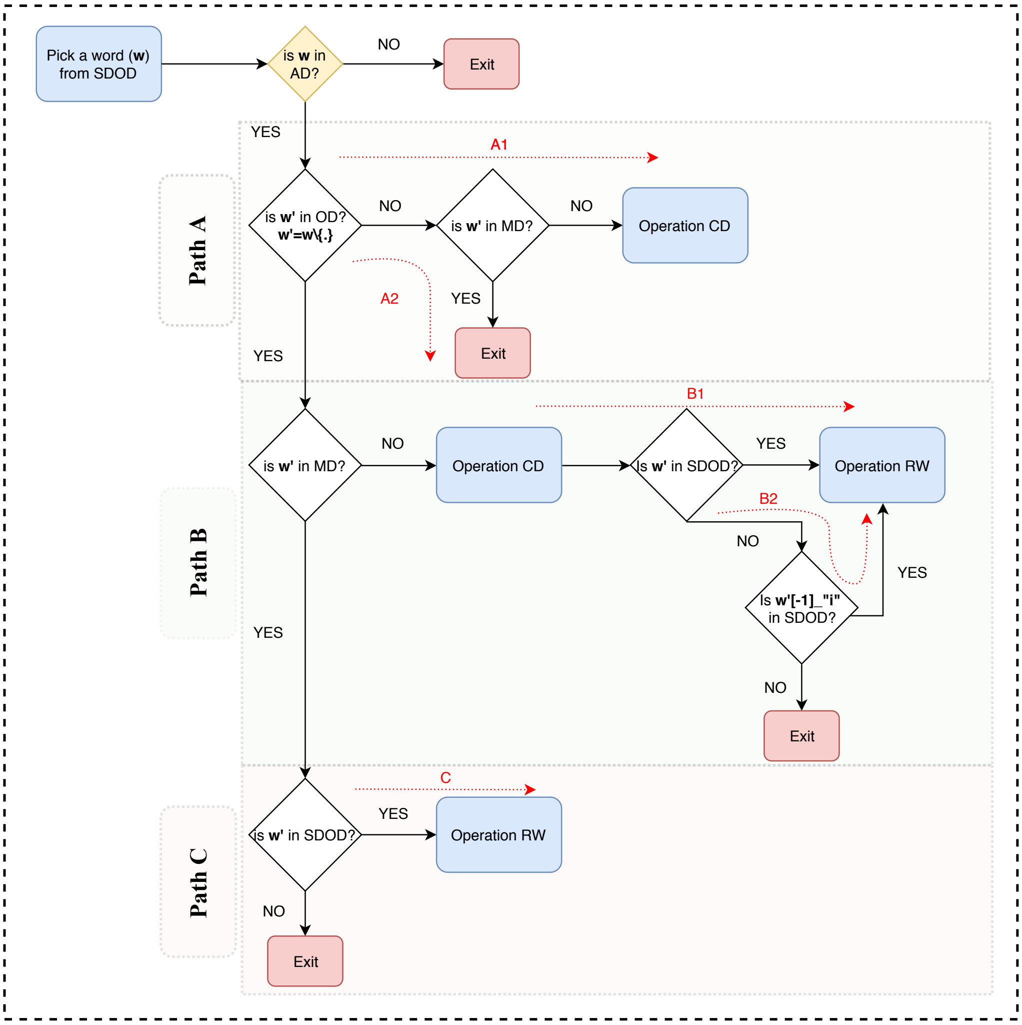


Figure 4. Flowchart represents the paths A, B, and C to update Levenshtein distance matrix.

In Figure 4, Levenshtein Matrix is derived by applying Levenshtein Distance algorithm (Figure 3) iteratively on the terms in original dictionary (OD). Other than original and universal dictionary we use a general medical dictionary (MD) as a simple filtration tool which exploits some available comprehensible medical terms out of meaningless words. Then, the flowchart modifies the Levenshtein matrix in order to improve the representativeness of the similarities between medical terms in unstructured text. Each path checks for the fulfillment of dictionary-related conditions (OD, MD, and SDOD) and leads to execution of two operations, removing words (RW) and changing distance (CD) (see Figure 5 and 6). Below we will explain the performance of these paths in detail.

- Path A: This path is for the abbreviation words of original term w assuming that OD has this abbreviation with a "." in the end. Therefore, CD operation in this path performs n-grams analysis to determine the initial or ending part truncation abbreviations.
- Path B: If there is a typo, then it will be the w' with a forgotten "."; this situation will be handled in Path B when we are scanning the words j (w' with forgotten "." will appear as a j candidate. This path is for the abbreviation-like words which can be caused in at least two scenarios: first is end of sentence period might make a word look like an abbreviation. Second is a typographic error where "." is entered by mistake after a word without a space. In such cases, this path B will correct it. It also adjusts the Levenshtein distance for abbreviations whose "." is forgotten (see description in Path A).
- Path C: While the other two paths, A and B, mainly process the words that does not exist in MD, this path applies RW operation on those terms exist in OD and MD but not in SDOD. Such words are full spelling medical terms that happened to be in abbreviation dictionary due to attaching to "." as ending punctuation in a sentence. In some cases, the staff may include more than one sentences in a single description which cause this ambiguity regarding the abbreviation detection when automating this process (see Figure 6).

3.2.3 Optimal Distance Weight

The proposed optimal weight for abbreviation and full spelling pairs distance is calculated by dividing each pairs distances by a lower bound of the distance values in Levenshtein matrix

Algorithm 3 Changing Distance Operation(t, SDOD, $P_{t,j}$, Lev_Matrix)

```
procedure CHANGING-DISTANCE($(t, j)$)
    $L_1 = Length(w)$
    $w = t \setminus "."$
    for each word $j$ in $SDOD \setminus w$ do
        $L_2 = Length(j)$
        if $L_1 > 1$ AND $L_2 \geq L_1$ then
            $n\_set = n\_grams(j, L_1)$
            $w\prime[-1]\_"i" = (w \setminus ".") + "i"$
            if $w = n\_set[0]$ then
                $Lev\_Matrix[\mathrm{t}, j] = P_{t,j} \times Lev\_Matrix[\mathrm{t}, j]$
            else if $w\prime[-1]\_"i" = n\_set[0]$ then
                $Lev\_Matrix[\mathrm{t}, j] = P_{t,j} \times Lev\_Matrix[\mathrm{t}, j]$
            end if
        end if
    end for
    Return $Lev\_Matrix$
end procedure
```

Figure 5. Changing distance of abbreviations and complete form and its derivatives.

(LB_D). The reason lies in the mechanism of the hierarchical clustering (with single linkage method) which groups the words (clusters) which share minimum distance value. Using equation eq:erl5 the weight values (P_{t,t_b}) can be found for each candidate pairs in order to modify the corresponding distances to local minima distance value in Levenshtein matrix. This guarantees the triggered merging task in initial step of the HAC implementation.

$$P_{t,t_b} = min(Lev\text{-}Matrix_i) \tag{1}$$

This optimal weight is used in operation CD to reduce the distances of candidate full spelling words and their abbreviations (see Figure 5).

Note that in CD operation, n-grams method plays an important role in identifying the sequence of characters in single words based on the n tokens triggered iteratively.

To validate the process demonstrated in the flowchart of Figure 4, we represent an example for each path per specialty in Table 1.

Algorithm 4 Removing Word Operation(t, SDOD, Lev_Matrix)

```
procedure REMOVING-WORD(($t$))
    $w = t \setminus "."$
    if $w \in SDOD$ then
        $Freq\_w = Freq\_t + Freq\_w$
        $SDOD \rightarrow SDOD \setminus t$
        Remove $Lev\_Matrix[\mathrm{t}, w]$
    else if $w'[-1]\_"i" \in SDOD$ then
        $Freq\_w'[-1]\_"i" = Freq\_t + Freq\_w'[-1]\_"i"$
        $SDOD \rightarrow SDOD \setminus t$
        Remove $Lev\_Matrix[\mathrm{t}, w'[-1]\_"i"]$
    end if
    Return $SDOD$, $Lev\_Matrix$
end procedure
```

Figure 6. Removing the words while duplication of their full spelling word as an abbreviation exist in SDOD.

Specialty / Path		***A***		***B***		***C***
		A_1	A_2	B_1	B_2	
Urology	***w***	app.	▨	cysto.	urethra.	cystoscopic.
	w′	app	▨	cysto	urethra	cystoscopic
	j	applic applica applicat applicati applicatio application	▨	cysto cystoscop cystoscopi cystoscopoy	urethral urethra urethr	
General	***w***	explor.	▨	lap.	laparotomy.	port.
	w′	lap	▨	lap	laparotomy	port
	j	explor explora explorat explorati exploratio exploration	▨	lap laparoscop laparoscopi laparoscopic laparoscopic laparoascopic laproscop laparscop	laparotomi	
OBGYN	***w***	pacu.	system.	condyloma.	cystoscopy.	assist.
	w′	pacu	system	condyloma	cystoscopi	assist
	j	pacu		condyloma	cystoscopi	assist
Cardio	***w***	rep.	area.	bypass.	biopsy.	vein.
	w′	rep	area	bypass	biopsi	vein
	j	replac replace replacem replaceme replacemen replacement		bypass bypasstim bypasstime bypasstimes	biopsie biopsies biopsis	
Other	***w***	bone.	none.	catheter.	exostectomy.	buttocks.
	w′	bone	none	catheter	exostectomi	buttocks
	j	bonea boneam boneamp		cathet cathete catheter	exostectomi	

Table 1. Examples for paths A, B, and C.

3.3 Hierarchical Agglomerative Clustering (HAC)

Clustering methods can be divided into two categories in terms of the grouping procedure, partitional and hierarchical. Partitional clustering such as k-means has been applied to many real world data recently and providing high quality clusters. The k-means algorithm is one of the most popular partitional clustering methods currently due to its running time and simplicity in practically understanding and deploying. Later on, various extensions of this method such as K-mediod, K-interval [36], K-mode [41] are designed to improve the outputs for designated areas of data mining and pattern recognition. However, still with these extensions in-use some hierarchical methods can outperform partitional methods such as k-means in medical text mining. One of the issues of partitional approaches such as k-means is that the number of clusters need to be determined at the beginning of the clustering. With lack of knowledge about large datasets with complex structures and possible oscillations in clusters it is hard to determine the optimal number of clusters.

However, extensive effort has been undertaken to find the answer to this research question but many found the right value of this parameter using heuristic approaches such as elbow method which is mostly applicable for specific data structures. While complications of finding initiative parameters such as optimal k (cluster numbers) is still there, hierarchical clustering methods are designed to eliminate the requirement of defining such inputs. One of these approaches is a greedy algorithm called Hierarchical Agglomerative Clustering. Other than simple and straightforward implementation, HAC provides more interpretive clusters in the shell of hierarchy structure compared to the unstructured flat clusters set derived by k-means. Accordingly, this helps in making decision on the level of clusters simply by analyzing the dendrogram levels.

In this study, we applied hierarchical agglomerative clustering with single-linkage method and euclidean linkage metric on our customized Levenshtein matrix (Lev-Matrix). HAC then provides a matrix (Z-Matrix) consisting information regarding each clustering iteration such as iteration number, grouped cluster IDs, distance value, and total number of words. More precisely, this bottom up approach merges the terms (or the clusters) at each stage based on minimum intra-cluster distances calculated by developed distance metric. Then, the dendrogram cutting levels can be defined based on the cluster size and minimum inter-cluster distance of the largest cluster for each cut depth in elbow graph. The lack of good theoretical and applied study in this regards, of course, makes it more challenging to reify the clusters. This has been broadly discussed in the next subsection.

3.4 Finding the best cutoff distance of the dendrogram

After obtaining the hierarchical representation of the clusters, in order to find the best dendrogram cutting depth we go beyond the current theories and heuristically uncover a new metric which helps reducing the ratio of false positive and false negative instances to the total number of unique words in the dataset for each cutting point. Previously, elbow graph has been used to determine the optimal number of clusters from dendrograms by the elbow location [98] but here we tend to look into more detailed information on how the clusters change per cut to decide which represents the target depth jump among the cuts more effectively by introducing a new evaluation metric for analysis in this graph. This metric can be used extensively for choosing the best cutoff point in the dendrograms empirically and eliminates human error which rises from human judgment as usually the investigators inspect the final dendrogram and select this point based on their own judgment. While the traditional metrics including silhoutee measure fail to find the best k clusters for the optimal distance depth (d), the presented measurement transforms the distances to more applicable gauge, namely cluster weight at cutoff point d or W_{C_d}, and reports the best cutoff value candidate at the first recognizable downward trend of the graph till the point where the second upward trend is observed.

In the following equation, we define the $Norm_{ML_d}$ as normalized value of the length of the longest cluster which has the highest average intra-cluster distance and $Norm_{S_d}$ as the normalized value of the total number of clusters in the clustering results at cutoff threshold (d). Additionally, $Mean_{MID_d}$ represents the average pairwise intra-cluster distance of the proposed cluster with $Norm_{ML_d}$ parameter. Consequently, $Mean_{MID_d}$ divided by 2 provides the mean intra-cluster distance of the word members without duplication. Apparently, as the length of largest cluster ($Norm_{ML_d}$) increases the total number of clusters decrease ($Norm_{S_d}$) since at smaller cutoff points more singleton clusters can be observed and at higher levels HAC method tries to merge these clusters in a bottom-up fashion. Moreover, we trigger the largest cluster of each cut which has the property of possessing the maximum intra-cluster distance because it highlights the suspicious case of an erroneous cluster merge step in the hierarchy.

Algorithm 5 Sample cutoffs S([LB,UB]) for Dendrogram D

```
procedure SAMPLE CUTOFFS(S(Z-Matrix, N))
    $S = \emptyset$
    $N = 10$
    $Step = 0$
    $UB = \lceil \min(\text{Z-Matrix.distance} \setminus \{0\}) \rceil$
    $LB = \lfloor \max(\text{Z-Matrix.distance} \setminus \{1\}) \rfloor$
    delta $= \frac{UB-LB}{N}$
    while $Step < UB$ do
        $Step += delta$
        $S = S \cup \{Step\}$
        Return S
    end while
end procedure
```

Figure 7. Generating sample of distances for cutting the dendrogram.

In HAC, increasing cutoff leads to reduced number of clusters (desirable) at the expense of increased intra-cluster distance (undesirable). Proposed formula trades off these changes and focuses on the longest cluster which dynamically changes with the cutoff distance. Numerator of this expression measures the size of the cluster (focus on the longest one) as a function of number of terms in cluster and average intra-cluster distance. The numerator increases with the increased cutoff distance. Denominator of this expression measures the number of clusters, i.e. increases with cutoff distance as the number of clusters decreases. Hence the ratio decreases if the rate of decrease in the number of clusters is greater than the rate of the increase in the size of clusters. We thus identify such regions where there is such a decrease in the ratio and at the lowest cutoff distance possible.

$$W_{C_d} = \frac{Norm_{ML_d} + (\frac{Mean_{MID_d}}{2})}{1 - Norm_{S_d}} \tag{2}$$

We will calculate the cluster weight using equation eq:erl for each cutoff point while a set (S) of cutoffs can be found from an evenly spaced samples computed over the interval with lower bound LB and upper bound UB. In Sample cutoffs algorithm, considering that the size of S is equal to 10, we can generate the sample cutoff set S given the Z-Matrix from HAC step (see Figure 7).

Algorithm 6 SelectionSort(C-Matrix[ID,V1,...,Vn-1])

```
Sorted-C=$\emptyset$
for $i = 0$ to $n - 2$ do
    min=i
    for $j = i + 1$ to $n - 2$ do
        if $C - Matrix.ID[j] < C - Matrix.ID[min]$ then $min = j$
        end if
        Swap C-Matrix[i] and C-Matrix[min]
    end for
end for
Return C-Matrix
```

Figure 8. Selection sort algorithm as an initiative step for HCHAC.

3.5 Heuristic Clustering of HAC (HCHAC)

One of the main issues of implementing HAC is that when cutting the dendrogram at a distance level some clusters may be splitted at the cut level with very small difference in terms of the similarity measure compared to the words that remained together. The reason lie in the minimum distances at each hierarchy and a predefined constant cutting level. Therefore, by specifying any cutting point some words may be pruned from the parent clusters. This problem rises some number of singleton cluster pairs or a singleton and a non-singleton cluster pairs as false negative (FN) samples. In this case, after performing first step of clustering the results show there exist some consecutive clusters that contain false negative cluster members, meaning that the words in two consecutive clusters represent the same word but not grouped in the same cluster. Consequently in the final phase of this study, Heuristic Clustering algorithm of HAC (HCHAC) is represented to reduce the FN instances derived from HAC clustering method by sorting the clusters based on their cluster ID. Algorithm represented in Figure 8 returns a sorted clustering matrix (C-Matrix) with respect to the cluster IDs.

C-Matrix provides the clusters information including cluster IDs and members. When sorting the matrix by cluster IDs, the aforementioned pattern can be perceived quite clearly. For example, HAC couldn't capture "lepp" as misspelled form of the word "leep" and match them at the designated threshold cut level. While the word "lepp" falls in cluster 67 and "leep" in cluster 68, this phase merges them into a single cluster. The second clustering algorithm, HCHAC, focuses on such instances and form a single cluster by merging two consecutive groups.

The main idea in this step is to decide merging based on some factors such as the length of exemplar word of the cluster where exemplars are selected based on the frequency of cluster terms in text. Moreover, whether the initial characters of the two candidate words match. The algorithmic representation of this step-wise approach is offered in algorithm described in Figure 9.

Algorithm 7 Heuristic Clustering of HAC(C-Matrix)

```
procedure HCHAC(C-Matrix[ID,M])
    $D = 0.3$
    $L = 4$
    for $i = 0$ to $n - 1$ do
        if $Length(M[i]) > 2$ AND $Length(M[i + 1]) > 2$ then
            if Levenshtein-Distance($M[i][0]$,$M[i+1][0]$) $\leq D$ AND Length($M[i][0]$) $> L$ then
                if $M[i][0 : L] = M[i + 1][0 : L]$ then
                    Merge C-Matrix[i] & C-Matrix[i+1]
                end if
            end if
        else
            $D = 0.25$
            if Levenshtein-Distance($M[i][0]$,$M[i + 1][0]$) $\leq D$ then
                if $M[i][0{:}2] = M[i + 1][0{:}2]$ then
                    Merge C-Matrix[i] & C-Matrix[i+1]
                end if
            end if
        end if
        end for
        Return C-Matrix
    end procedure
```

Figure 9. HCHAC algorithm for empowering HAC.

In the procedure above the length of the first word in each cluster is a determinant parameter in emendation of false negative cluster members. By optimizing the thresholds we can assume that the words with 3 or 4 characters are considered to be short medical terms for which the rule of identical first 2 characters apply. Also the words with length of more than 4 tend to be merged into single cluster if the initial 3 characters match according to Figure 9. Also the optimized distance threshold of the exemplar words is 0.25 and 0.3, respectively. Therefore, HCHAC can be introduced as a spell checking process which is clustering the initially formed clusters based on words length, comparison of first portion and Lenvenshtein distance of the words. Some HCHAC singleton and non-singleton clusters merging examples from Cardio specialty are presented in Table 2. Evidently, applying HCHAC as a complementary step for HAC method is more informative rather than misleading. This can be observed in all specialties multi-step clustering results.

Cluster ID	Words		Result
129	roon	-	1 FP
130	room	robot	
248	laparoscopi	-	1 FP
249	laparoscop	laparotomi	
262	mini	minim	2 TP
263	min	-	
332	extend	extens	2 TP
333	externd	-	

(a) Table 2a. HCHAC singleton clusters candidate from Cardio specialty.

Cluster ID	Words	Result
17	aneursym	1 TP
18	aneurysm	
29	dddr	1 TP
30	ddd	
39	nmud	1 TP
40	nmuc	
48	mosaic	1 TP
49	mosiac	
59	hrs.	1 TP
60	hr.	
94	site	1 FP
95	side	
159	drainag	1 TP
160	drainage.	
186	repiar	1 TP
187	repair	
245	endograph	1 TP
246	endograf	
255	ligatur	1 TP
256	ligat	

(b) Table 2b. HCHAC non-singleton clusters candidate from Cardio specialty.

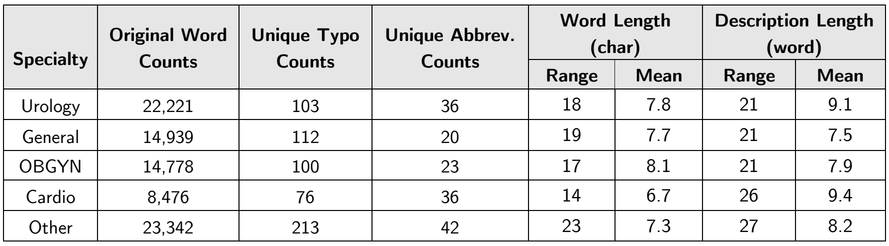
Table 2. HCHAC complementary step for clusters grouping. In Table 2a you can see that 2 FPs and 4 TPs and in Table 2b 1 FP and 9 TPs are added to the results of HAC. Therefore, the gain of this method is 13 TPs as opposed to 3 FPs loss.

Output	Dictionary (Universal, Original)		Distance Matrix (Lev-Matrix)		Cluster Matrix (Z-Matrix, C-Matrix)		
Step	1. Preprocessing	2. Add Stem Derivatives	3. Generate Levenshtein	4. Update Levenshtein	5. Run HAC	6. Selection Sort (HAC clusters)	7. Run HCHAC
Key Task(s)	• Remove special char • Remove stopword • Tokenization • Stemming	• Add derivative terms • Add frequencies	• Calculate Levenshtein matrix	• Scenario 1: • Remove "t_b" • Freq(t) = Freq(t_b) + Freq(t) • Scenario 2: • Find: • First part shortenings • Last part shortenings • Optimal penalty w_i	• Single linkage • Provide pairwise merging steps details	• Sort the clusters based on cluster IDs	• Merge C[i], C[i+1] if: • Distance(M[i],M[i+1]) ≤ 0.25 • Length(M[i])>4 & first 3 chars match • Length(M[i])$\leq$ 4 & first 2 chars match

Table 3. Study Framework and steps of this research along with methodologies and outputs.

3.6 Medical Text Analysis Results

In our evaluation of the presented approach, we used historical surgery datasets. The historical datasets were extracted from a hospital's surgery database spanning a three-year period. The raw text fields extracted include free text surgical notes and procedure descriptions of the surgeries entered manually by the surgery department's staff and clinics. The surgery description corpus contains in total 17% misspellings and 8% abbreviation words. Given that misspellings and abbreviations depend on the specialty, the data set is organized into five main specialties each with sufficiently large number of observations. The last specialty, called "Other", is an aggregation of all other specialties with fewer observations. Table 3 and second and third columns of Table 4 present the detailed dataset information for each specialty. We note that there are more misspelled terms than the abbreviations in all specialties.



Specialty	Original Word Counts	Unique Typo Counts	Unique Abbrev. Counts	Word Length (char)		Description Length (word)	
				Range	Mean	Range	Mean
Urology	22,221	103	36	18	7.8	21	9.1
General	14,939	112	20	19	7.7	21	7.5
OBGYN	14,778	100	23	17	8.1	21	7.9
Cardio	8,476	76	36	14	6.7	26	9.4
Other	23,342	213	42	23	7.3	27	8.2

Table 4. Insights about text data per specialty

3.6.1 Obtaining Ground Truth

The ground truth in this study corresponds to the correct clustering of misspelled variants and abbreviations of each term (medical or non-medical) of the corpus. In finding ground truth, we followed a manual process of labeling by human experts which are inherently subjective, and thus cannot be expected to be 100% accurate but rather expected to be informed. To reduce the inherent subjectivity and increase the accuracy, we have performed wildcard pattern searching using medical dictionaries, CPT databases, and UMLS. To illustrate, consider the cluster {"laparoscopi","laparoscop","laparotomi"} obtained from HCHAC process. We do 3 wildcard searches using the dictionaries would result in "laparotomi" to be identified as the stem of "laparotomy" which is different than the "laparoscopy". Another cluster example is {"roon", "room", "robot"} for which manual processing identifies "robot" to be separated from the rest.

To illustrate a case for the limitation of this manual labeling approach, consider the cluster {"contert", "content", "convert"} where "contert" can be the typo with original form of either "content" or "convert". To decide which cluster this typo belongs to, we further considered the context of the term and co-occurrence similarities with other terms in the raw data set. In such ambiguous cases where it was not possible to determine the correct cluster assignment, we created scenarios of multiple ground truth cluster assignments and report our results based on these scenarios.

Creating ground truth clusters from scratch is a very time-consuming task (i.e., pairwise comparison of all terms) due to difficulty level of medical terms and size of the word dictionary. Instead, we begin the manual ground truth discovery process with the cluster outputs of HCHAC method for each specialty. This initialization has significantly reduced the manual labeling effort and allowed for wildcard pattern search with multiple sources.

3.6.2 Impact of Clustering Method

To better judge our model's capability in standardizing the free text descriptions, we first compared the performance of the HAC (our selected clustering approach) to that of Density-based spatial clustering of applications with noise (DBSCAN)[24]. The results of reduced text features in standardized text using HAC and DBSCAN are presented in Table 5. Both HAC and HCHAC methods are able to reduce number of text features (i.e., clusters of words) by identifying misspellings and abbreviations. When compared with stemming-based reduction, HCHAC is able to detect 17% to 37% of the terms as misspellings and abbreviations with an average of 26% across all specialties. While the DBSCAN is able to reduce number of clusters more than HAC and obtain a better misspelling detection accuracy, its clustering and abbreviation accuracies are worse. We used the following expression to calculate the accuracy and the ground truth (explained next).

In an effort to improve the performance of DBSCAN, we applied the DBSCAN to the clusters obtained from HAC, i.e. HAC-DBSCAN, which slightly improved the accuracy results. The two-step approach in Algorithm 6, HCHAC, displays best performance in terms of all accuracy types.

3.6.3 HAC and HCHAC Clustering Results

The output of the presented approach is a clustering of the terms in the corpus, i.e., misspellings, abbreviations, full spelling of words, correctly spelled words. The underlying hypothesis in generating clusters of terms is that each cluster consists of a correctly and fully spelled word, its misspelled and/or abbreviated variants. In order to test the performance of the proposed approach's cluster results of misspellings and abbreviations, we compared the results with the ground truth results obtained by manual processing. We note that manually assigned category labels are usually used as a diagnostic baseline criteria for the evaluation of clustering results. With this kind of evaluation, we are intrinsically assuming that the objective of clustering is to replicate human thinking and processing. We report on the misspelling and abbreviation detection accuracy, as well as clustering precision, recall, accuracy and F1-score based on their consistency with the manually created ground truth (gold standard clusters).

Assessing precision, recall, and F1-score, accounts for the clustering and the ground truth cluster assignments as comparable groups explained by the particular TP, FP, and FN instances. The precision and recall scores of clusters in clustering results, C, and their equivalent ground truth clusters, M, are obtained by finding instance pairs of cluster members in both C and M based on the criteria represented in the following confusion matrix (Table 6).

The precision and recall and accuracy are computed by equations (7), (8), and (9) in the previous section. F1-score is the harmonic mean of these two measurements. [4]

Specialty	After Preprocessing	After Stemming	HAC Clusters	HC-HAC Clusters	DBSCAN Clusters	HAC-DBSCAN Clusters
Urology	862	663	435	415	423	405
General	784	653	545	515	490	494
OBGYN	809	667	471	449	433	438
Cardio	761	596	521	494	477	477
Other	1,579	1,271	1,017	973	889	892

Text feature reduction while standardizing the free text

Average	Clustering Accuracy	TP Rate	Typo Accuracy	Abbrev. Accuracy
HAC	80.2	86.4	74.2	80.4
HC-HAC	84.4	90	90.6	90
DBSCAN	66.2	70.8	79.2	75.4
HAC-DBSCAN	68.2	73.6	79.2	80.2

The performance measure (average) for different models

Table 5. Feature reduction in each specialty with respect to the performance measure

# of instances	True	False
Positive	$\in$ C & $\in$ M	$\in$ C & $\notin$ M
Negative	$\notin$ C & $\notin$ M	$\notin$ C & $\in$ M

Table 6. Confusion matrix in clustering

Figures 10 - 13 represent the precision, recall, F1-score and accuracy percentages respectively for HAC and HCHAC methods under two scenarios of the study; HAC worst, HAC best, HCHAC worst, and HCHAC best. Here '-best' and '-worst' denotes the best and worst performance level obtained across all scenarios of ambiguous case cluster assignments. Results in Figures 10 - 13 show that while HAC results are promising, HCHAC is able to improve over HAC results further, especially in terms of accuracy. For instance, accuracy and F1-score percentage of HCHAC lies in [91,92] and [85,87], respectively, which corresponds to almost 8% average improvement in clustering results.

To better assess these results, we compared with a baseline approach. Since HCHAC is an unsupervised approach and does not require a reference source (e.g., dictionary) and there exists no other study which does not use a reference or a dictionary, we are unable to provide a comparable baseline approach from the literature. Instead, we provide the results of a simplified approach which uses parts of the presented methodology, i.e., updating LD for abbreviations and optimal threshold setting. The results of this baseline is presented in Table 5 compared with HCHAC. Results show 8-10% average accuracy improvement over the baseline established. Lastly, we also evaluated the effect of the size of the corpus on the stability with respect to three different accuracies. Our experiments showed that, while different accuracies (clustering, typo, abbreviation) stabilize at different corpus sizes across specialties, the stability of the clustering model (HCHAC) is attained by using at least 75% of each specialty's corpus.

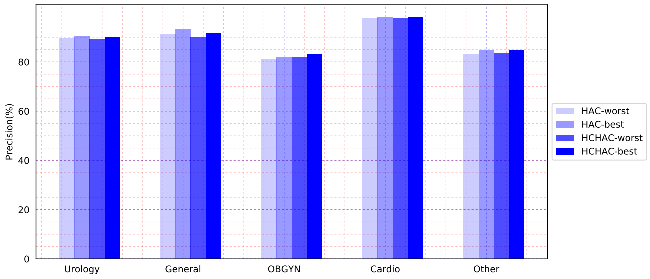


Figure 10. HCHAC and HAC precision (worst and best scenarios) plot.

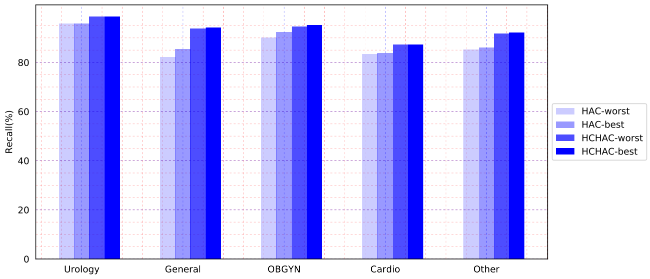


Figure 11. HCHAC and HAC recall (worst and best scenarios) plot.

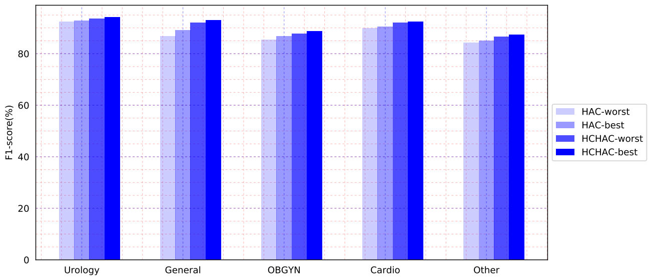


Figure 12. HCHAC and HAC F1-score (worst and best scenarios) plot.

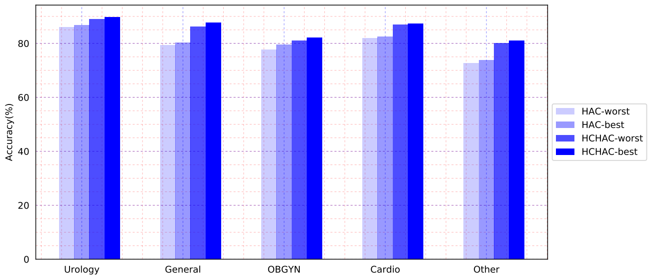
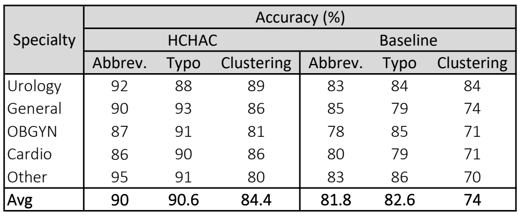


Figure 13. HCHAC and HAC accuracy (worst and best) scenarios plot.



Specialty	Accuracy (%)					
	HCHAC			Baseline		
	Abbrev.	Typo	Clustering	Abbrev.	Typo	Clustering
Urology	92	88	89	83	84	84
General	90	93	86	85	79	74
OBGYN	87	91	81	78	85	71
Cardio	86	90	86	80	79	71
Other	95	91	80	83	86	70
Avg	90	90.6	84.4	81.8	82.6	74

Table 7. The accuracy of abbreviation and typo detection in text

3.6.4 Choosing the best cutoff point

As discussed in the methods section, we developed a new metric in elbow method to find the best cutoff point in the generated dendrogram. We illustrate the effectiveness of this approach using the historical data set. In Figure 14, the x-axis shows the range of cut-off values for each specialty. We describe a specific range of cut-off points for each specialty based on the minimum and maximum distances in dendrogram as described in Algorithm 4. The candidate thresholds for Cardio, General, OBGYN, Urology, and Other specialties are (0.16, 0.18), (0.19, 0.22), (0.19, 0.21), 0.18, and 0.17 respectively. Note that some specialties have multiple cut-off points in the region of interest, i.e. where the metric is decreasing first time.

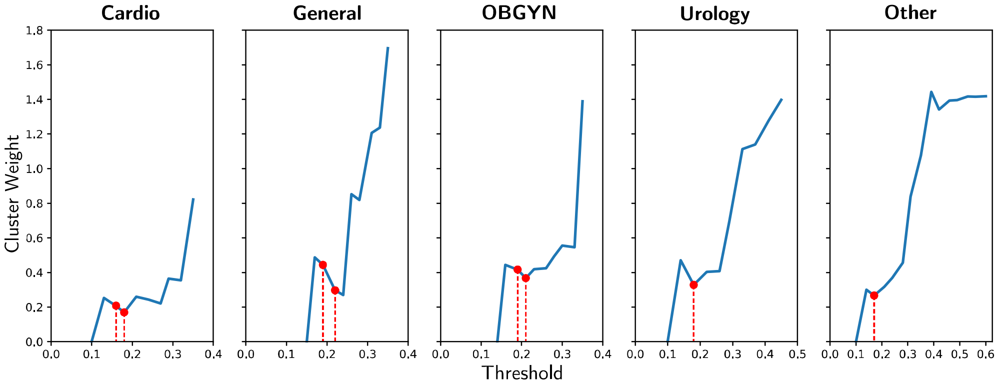


Figure 14. Cluster Weights on different random dendrogram cut for each specialty.

In order to assess the quality of the identified thresholds for cutting the dendrogram (for each specialty), we experimented with different cut-off levels and report the precision, recall, and F1-score of the clustering results at each cutoff level (Figure 15). According to these results, the optimal cuts are indeed the ones identified with our proposed cut-off threshold procedure. For the specialties where there are multiple cut-off candidates, Figure 15 confirms that one of the candidate cut-off levels is the optimal level. For instance, the optimal cut-off point for "Cardio" is identified to be either 0.16 or 0.18. Referring to Figure 15, the precision is maximized at 0.18 with an acceptable recall percentage, 83.3%, which results in highest F1-score percentage

(90%) among all cutoff points for this specialty.

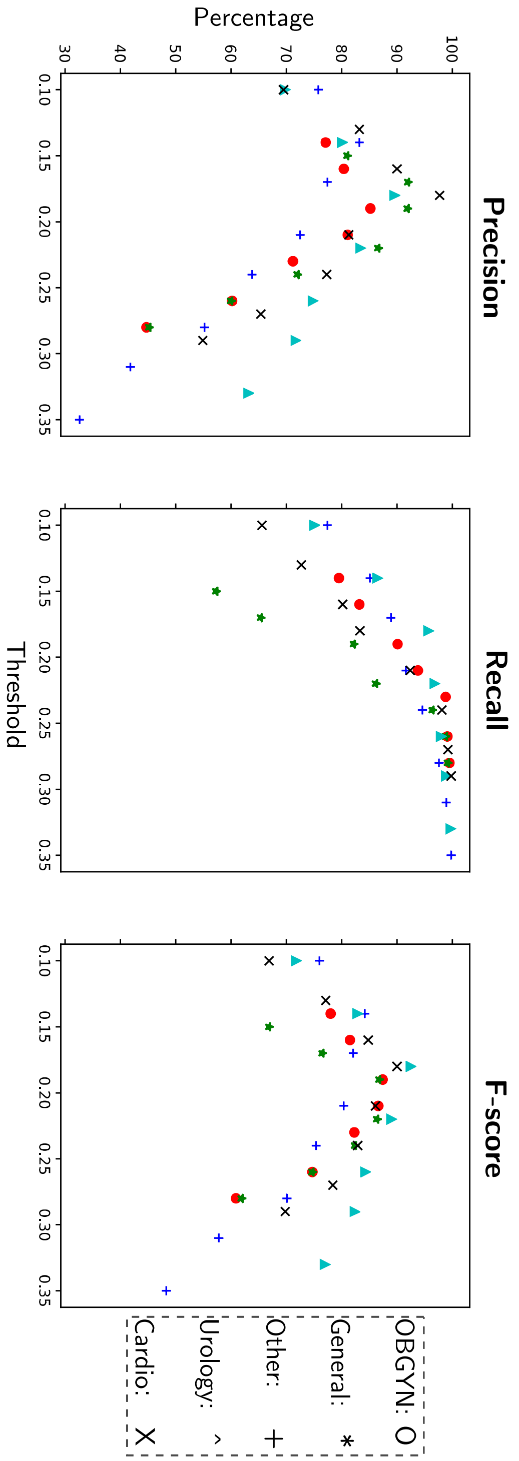


Figure 15. P/R/F for different cuts per specialty.

4 Classification: Primary and Multiple CPT Prediction

While some surgical units use preliminary CPT codes for operational scheduling, majority of services choose to only use textual descriptions and key identifiers. Identifying key sub-procedures and their importance in estimating the surgery duration are usually based on the schedulers' knowledge and years of experience. Even those providers using preliminary CPT codes end up changing their CPTs upon the completion of the surgery by examining the surgery notes and outcomes (i.e., medical coding for billing). Hence, knowing the accurate CPT code(s) for a surgery case in advance is critical for operations planning such as scheduling and equipment provisioning. Therefore, in this chapter, we develop multi-class classification models and enhancements to predict the CPT codes and improve the outcomes. The presented models use the structured text data from the previous chapter to improve the feature set for predicting more accurate CPT codes.

4.1 Primary CPT Prediction

- Data Collection

  Data should be collected from interactive information system's relations in surgical units. The consistency of the data determines how close the prediction will be to the ground truth labels. We have determined the most influential inputs for this predication task. These inputs are surgeon ID, patient's age, case type, TFIDF (term frequency score of text data processed in the previous step) and surgery scheduled duration. The dataset contains about 28,000 of records of surgery scheduled cases, containing operations details in the period of May 2013 till June 2017 in the main OR suite. However, after performing initial data cleaning, the size of the dataset is reduced to 10,000 cases. Also, the feature set of the data that is used in the analytical pipeline of this research includes both categorical and continuous variables.

- Preprocessing

  We have extracted the importance scores of the words in procedure descriptions after applying our text mining research plan [47]. Moreover, prior to any prediction effort, we have investigated whether actual surgery durations follow a well-known parametric distribution, such as normal or log normal. This will help us to determine if we can use a simple parametric technique for estimation or we need to propose a method that look into the parameter relations and prediction task more precisely [44].

- Prediction Models

  In surgical CPT prediction studies, surgery description holds invaluable information [58]. The text entry in the system of many hospitals is pattern-free as surgery descriptions in surgical records. Therefore, any prediction attempt directly from unprocessed text simply results in poor estimations with high prediction error. In the previous section, we have identified structured feature set by utilizing an unsupervised text mining approach from the free-text descriptions [47]. Current literature failed to use this important feature in CPT prediction task however we think it improves the accuracy of the prediction model considerably. The ongoing and standard CPT assignment approach involves manual labeling, which entails significant human effort and is cumbersome for huge surgery schedule databases in large hospitals.

  In our data set, CPT codes are not provided as a singleton, rather as a set of CPT candidates. In other words, the surgeries are represented in terms of single or in many cases multiple CPT codes; that is named "CPT list". This is because most surgeries have multiple components, some of which are standard procedures such as anesthesia etc. and others are multiple procedures being done concurrently. The class assignments, using either of these two methods, are the predicted singleton CPT codes for each surgery. The single CPT label reflects the most dominant procedure among other CPTs. To train the model with single label, we have extracted the dominant single CPT from the CPT set by reviewing their corresponding Relative Value Unit scores. Relative Value Units (RVUs) are the effort to quantify the physician services for reimbursement in US Medicare system. The CPT which owns the highest RVU score is chosen as the dominant CPT in the surgery CPT set; that is named "primary CPT" [54].

  Subsequently in this discussion, surgical CPT codes are predicted through supervised methods, namely Random Forest (RF) [12] or Extreme Gradient Boosting (XGBoost) [15] classifiers. We mention both models as effective methods for modeling multi-class problems, yet both are flexible with many variable modification capabilities; e.g. loss function. These methods are well-supported by making decisions based on constructed probability tree of the possible scenarios and outputting the class label which is the mode

```
Algorithm 8 RF($S_{train}$={($x_1$,$y_1$),...,($x_N$,$y_N$)}, $A = \{1, ..., N\}$, $t = \{1, ..., ntree\}$)
 1: procedure RF($S_{train}$,$A$,$t$)
 2:     Bootstrap sampling $\leftarrow B_t$
 3:     $B_t \subset A$
 4:     $B'_t \leftarrow A\setminus B_t$
 5:     Fit classification tree to $B_t$
 6:     Obtain the predicted class of each terminal node
 7:     for each member $m$ in $B'_t$ do
 8:         Pass $m$ down the tree $t$
 9:         Terminal nodes $\leftarrow q_t(i)$ for $i \in A$
10:     end for
11:     Find frequency of the classes for observations $\in S_{train}$ of all trees in set $t$ for which the
    observation $\in B'_t \leftarrow$F$_m^{class \in c}$
12:     Class label of $m \leftarrow$ max(F$_m^{class}$)
13: end procedure
```

Figure 16. Random Forest classification algorithm.

of the leaves class assignments. While these methods are both ensemble learning models and using decision trees as the base learner, GB tends to use weak learners (shallow trees that could be as small as stumps) iteratively. On the other hand, unlike GB, RF uses fully-grown trees in parallel. We define RF and GB classifiers are represented as algorithms in Figures 16 and 17.

In RF algorithm, bootstrap samples are taken N times randomly with replacement at tree t. Moreover, in line 5 the classification process is acquired by splitting tree nodes of tree t on predictor variables, converging when reaching terminal nodes of the same class. In line 4, the class assignment of B_t members in each node of tree t is obtained.The optimal class for observations in test set is the label with maximum frequency of the same training observations at terminal nodes of all constructed trees. The standard method for probability estimation is based on the proportion of trees that predict class c when m is passed down the tree t (See lines 8-12 of algorithm 8).

In Figure 17, at each iteration of XGB model $F_{CM}(x_i)$ is computed by fitting a base learner to the negative gradient of the loss function L in regards to preceding iteration's output, $F_{c,m-1}(x)$. In step 7, it tends to train using (x_i, r_{icm}) for all $i \in \{1, ..., N\}$, iteration m, and class c. In line 9, one-dimensional optimization problem, $\gamma_{jcm} = argmin \sum_1^n L(y_{ic}, F_{c,m-1}(x_i) + \gamma h_{cm}(x_i))$, is solved for J terminal nodes to obtain the

Algorithm 9 GB(S_{train}=$\{(x_1,y_1),...,(x_N,y_N)\}$, $L = l(y, F(x))$, $n_{iteration} = M$, $c = \{1, ..., C\}$)

```
procedure GB($S_{train}$,$L$,$n_{iteration}$)
    Initialize model with a constant value: $F_{c0}(x) = argmin \sum_1^n L(y_i, \gamma)$
    for each iteration $m$ in $M$ do
        Compute $p_c(x) = \frac{exp(F_c(x))}{\sum_{l=1}^{c} exp(F_l(x))}$
    end for
    for each iteration $c$ in $C$ do
        Compute residuals: $r_{jcm} = -\left[\frac{\partial l(y_i c, F_c(x_i))}{\partial F_c(x_i)}\right]_{F_c(x)=F_{c,m-1}(x)}$ ∀j∈ $\{1, ..., J\}$
        Fit a base learner $h_{cm}(x)$ to $r_{icm}$
        Compute multiplier $\gamma_{jcm}$: $\gamma_{jcm} = \frac{C-1}{C} \frac{\sum x_i \in r_{jcm}(y_{ic} - p_c(x_i))}{\sum x_i \in r_{jcm}(|y_{ic} - p_c(x_i)|)(1 - |y_{ic} - p_c(x_i)|)}$
        Update the model: $F_{cm}(x) = F_{c,m-1}(x) + \sum_1^J \gamma_{jcm} h_{cm}(x)$ $(x \in r_{jcm})$
    end for
    return $F_{CM}(x)$
end procedure
```

Figure 17. Extreme Gradient Boosting algorithm for multi-class prediction [15, 29].

prediction in m_{th} iteration for class c. In multi-class case, for each pair of data (x, y), $y_c = 1$ is considered to be the class label of observation x while $y_c \in \{0, 1\}$, and we investigate $p_c(x) = p(y_c = 1|x)$.

The estimations in $F_{CM}(x)$ then can offer us the probability estimation $(p_{cM}(x))$ of all c classes in XGB model. The optimal parameters of the models can be obtained by cross-validated grid search over a given grid of parameters to iterate over different parameter value combinations. The parameter grid offers variations of important tree parameters such as minimum number of samples at leaves ($min\_samples\_leaf$), minimum number of samples to split an internal node ($min\_samples\_split$), and number of trees in RF or number of boosting levels to execute in XGB ($n\_estimators$).

- Class Weight Recalculation

We developed this method to apply it to the ensemble methods' output for reducing the noise caused by those data points which misinform the learner. As an instance, CPTs of similar procedures are different codes but their textual feature contents are similar with minor word differences. These divergences are muted and not resolved well in the ensemble methods hence it brightens the need for differentiating such observations with clarity. One solution is to design a CPT selection approach as a wrapper method which assigns weights to the most probable CPT sets. These CPT sets are in consistency with the most influential features, such as special characteristics of textual feature, model's feature importance measurements, and probabilities of the most possible CPTs. For building the wrapper, we need a model that informs the importance of text features and CPT assignment probabilities with respect to the classification task at hand.

In this approach, we used the RF's feature importance as one of the effective factors. RF calculates the feature importance measure based on calculation of the Gini impurity at each split node [6]. It worth to mention that XGB also produces potentially useful feature importance. In XGB model, feature importance is only defined if the decision tree model is selected as a base learner. However, these measures usually roll up with their own pitfalls mainly in data interpretation efforts. With correlated attributes in the feature set, potent features in prediction turn out to be less important based on assigned scores. In other words, such importance measurements can be biased towards variables with more categories. Instead using these importance weights in coding the input text features and retraining the model, we make perturbations of the probabilities of the predicted CPT alternatives.

Following the single CPT predicting task, we present a novel perturbation-based approach to improve the accuracy of prediction using the class probabilities extracted from RF probability matrix ($p_c(x)$ of all c classes given each surgery case x). The prediction probabilities of the alternative CPT classes (c) is then recalculated through a weighting scheme. Here, the probabilities of the CPT prediction alternatives would be altered resulting in the modified ordering of the label predictions with respect to the primacy of the CPTs in the class sequences based on calculated weights. For the ordered list of three CPTs with highest probabilities per surgery case, we represent CPT_{p_i} where $i = \{1, 2, 3\}$ and $i = 1$ denotes highest probability CPT and $i = 3$ denotes lowest probability CPT, then we have:

$$CPT_{p_i n} = [CPT_{p_1}, CPT_{p_2}, CPT_{p_3}]_{n \in \{1,\dots,N\}} \, \forall i \in \{1, 2, 3\} \tag{3}$$

Let $CPT\_catalog\_dict_{p_i}$ be the dictionary of tuples, $[(W_1, F_1, I_1), ..., (W_k, F_k, I_k)]$ for k words in CPT_{p_i} description word list, where $W, F, andI$ represents word in CPT description, frequency of W, and importance measure of W, respectively. Then we have:

$$CPT\_catalog\_dict_{p_i} = \{CPT_{p_i} : \{(W_1, F_1, I_1), ...$$
$$, (W_k, F_k, I_k)\}_{k \in CPT_{p_i}} \forall i \in \{1, 2, 3\}\}\} \tag{eqn}$$

In addition, let $CPT\_catalog\_dict_{actual}$ be the dictionary of tuples, $[(W_{n1}, F_{n1}, I_{n1}), ...$ $, (W_{ne}, F_{ne}, I_{ne})]_{n \in \{1,...,N\}}$ for e words in CPT_n description word list. Then we have:

$$CPT\_catalog\_dict_{actual} = \{CPT_n : \{(W_1, F_1, I_1), ...$$
$$, (W_e, F_e, I_e)\}_{e \in CPT_n} \forall n \in \{1, ..., N\}\}\} \tag{eqn1}$$

The weighting approach supports the improvement of CPT prediction accuracy by incorporating pairwise similarity (S), word frequency (F), and word importance measure (I). Algorithm 10 and 11 in Figures 12 and 13 are presented to calculate the class weights based on given variables. The coexistence of words in $CPT\_catalog\_dict_{actual}$ and $CPT\_catalog\_dict_{p_i}$ determines which algorithm should be used to compute the new weights for 3 most probable CPT assignments. The ultimate weight is calculated by following relational formula:

$$W = function(S_{w,w'}, F_w, I_w, F_{w'}, I_{w'}) \tag{6}$$

The magnitude of weight increases if the importance measure and frequency increases, $W \propto I \times F$. The reason is that if the frequency of a word feature in the learning model increases, the gini impurity measure of this feature increases too since it provides more classes in gini computation compared to a less frequent feature. Additionally, pairwise word similarity measure can greatly improve the weights as the co-occurrence of the medical terms in the candidate CPT and actual label descriptions reflects the level of both CPTs' procedure similarity.

Algorithm 10 $Weight\_Calc_1$ $(CPT\_catalog\_dict_{p_i}, CPT\_catalog\_dict_{actual}, CPT_n)$

```
1: procedure $Weight\_Calc_1(CPT\_catalog\_dict_{p_i}, CPT\_catalog\_dict_{actual})$
2:     $S \leftarrow 1$
3:     $F_{CPT_{p_{ik}}} = Count(w) \mid w \in CPT\_catalog\_dict_{p_i}[CPT_{p_i}]$
4:     $I_{CPT_{p_{ik}}} = Importance(w) \mid w \in CPT\_catalog\_dict_{p_i}[CPT_{p_i}]_k$
5:     $F_{CPT_n} = Count(w') \mid w' \in CPT\_catalog\_dict_{actual}[CPT_n]$
6:     $I_{CPT_n} = Importance(w') \mid w' \in CPT\_catalog\_dict_{actual}[CPT_n]$
7:     $W_{p_{ik}} = S \times (F_{CPT_{p_{ik}}}) \times (I_{CPT_{p_{ik}}}) \times (F_{CPT_n}) \times (I_{CPT_n})$
8:     $Assigned\_Weight \leftarrow W_{p_{ik}}$
9:     return $Assigned\_Weight$
10: end procedure
```

Figure 18. Algorithm for calculating class weight when word w is in both dictionaries, $CPT\_catalog\_dict_{p_i}$ and $CPT\_catalog\_dict_{actual}$.

Algorithm 10 is repeated for each high-priority CPT in $CPT\_catalog\_dict_{p_i}$ $(i = \{1, 2, 3\})$ and each observation in $CPT\_catalog\_dict_{actual}$ (n={1,...,N}) until the new weights for three high-priority CPTs of each surgery case is calculated. With N observations in surgery schedule, this algorithm outputs $I \times N$ matrix consisting of 3 weights for each observation. Lines 3 and 5 are given to compute the frequency count of the words w and w' in $CPT\_catalog\_dict_{p_i}$ and $CPT\_catalog\_dict_{actual}$ dictionaries, namely $F_{CPT_{p_{ik}}}$ and F_{CPT_n}, respectively (Note that $w = w'$). Additionally, in lines 4 and 6 the importance measure of these words are extracted from importance matrix of RF/XGB models.

Algorithm 11 (Figure 19) reflects the same behaviour with this difference: $w \neq w'$. In the light of this contrast, we can claim that $I_{CPT_{p_{ik}}} \neq I_{CPT_n}$. The Levenshtein distance [39] is computed to find the word pairs with maximum similarity measure and consider them in weight calculation. The $Thresh_Dist$ parameter is a threshold distance measure defined specifically for each specialty based on the level of dissimilarity in description words.

We illustrate the framework of our weigh assignment approach in Figure 20. The feature set is reprocessed [47] and the CPT labels are assigned to each case using RVU measures.

Algorithm 11 $Weight\_Calc_2(CPT\_catalog\_dict_{p_i}, CPT\_catalog\_dict_{actual}, Thresh\_Dist, CPT_n)$

```
procedure $Weight\_Calc_2(CPT\_catalog\_dict_{p_i}, CPT\_catalog\_dict_{actual}, Thresh\_Dist)$
    $Set_S \leftarrow \emptyset$
    for k tuples in $CPT_{p_i}$ do
        $S = 1 - Lev\_Dist(w \in CPT\_catalog\_dict_{p_i}[CPT_{pi}]_k, w' \in CPT\_catalog\_dict_{actual}[CPT_n])$
        Add $S$ to $Set_S$
    end for
    $S_{max} \leftarrow max(Set_S)$
    if $S_{max} > 1 - Thresh\_Dist$ then
        $F_{CPTp_{ik}} = Count(w) \mid w \in CPT\_catalog\_dict_{p_i}[CPT_{p_i}]$
        $I_{CPTp_{ik}} = Importance(w) \mid w \in CPT\_catalog\_dict_{p_i}[CPT_{p_i}]_k$
        $F_{CPT_n} = Count(w') \mid w' \in CPT\_catalog\_dict_{actual}[CPT_n]$
        $I_{CPT_n} = Importance(w') \mid w' \in CPT\_catalog\_dict_{actual}[CPT_n]$
        $W_{p_{ik}} = S \times (F_{CPTp_{ik}}) \times (I_{CPTp_{ik}}) \times (F_{CPT_n}) \times (I_{CPT_n})$
    else
        $W_{pik} = (S^2) \times (F_{CPTp_{ik}}) \times (I_{CPTp_{ik}}) \times (F_{CPT_n}) \times (I_{CPT_n})$
    end if
        $Assigned\_Weight \leftarrow W_{p_{ik}}$
        return $Assigned\_Weight$
    end procedure
```

Figure 19. Algorithm for calculating class weight when word w exists in $CPT\_catalog\_dict_{p_i}$ but not in $CPT\_catalog\_dict_{actual}$.

W take 80% of the entire data set of each specialty for training the classification model and obtaining two dictionaries: $CPT\_catalog\_dict_{p_i}$ and $CPT\_catalog\_dict_{actual}$. The model is tuned using grid search technique to find the optimal combination of hyper parameters. Given the CPT probability matrix, feature importance matrix (outputs of the fitted model), and training data, we extract importance measure and occurrence frequency of the words in the descriptions for 3 CPT assignments with highest probabilities and the actual CPT label. The framework presented in Figure 20 executes two algorithms given the word co-occurrence states. We repeat this process for each of the three CPT codes in $CPT\_catalog\_dict_{p_i}$ dictionary and obtain their new weights. Given the calculated weights, we can determine a new order of the high-priority CPT codes and offer a CPT assignment for each surgery case which may be different from the single CPT prediction output.

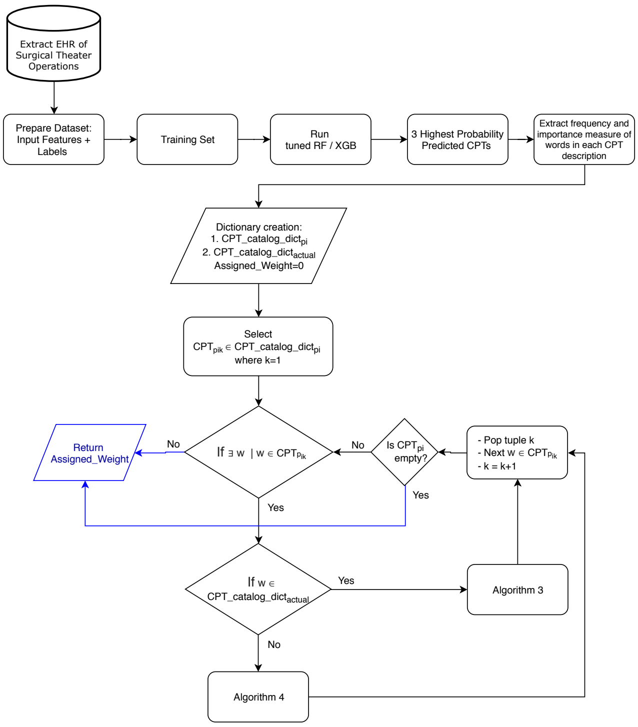

Extract EHR of Surgical Theater Operations
Prepare Dataset: Input Features + Labels
Training Set
Run tuned RF / XGB
3 Highest Probability Predicted CPTs
Extract frequency and importance measure of words in each CPT description
Dictionary creation: 1. CPT_catalog_dict$_{pi}$ 2. CPT_catalog_dict$_{actual}$ Assigned_Weight=0
Select CPT$_{pik}$ ∈ CPT_catalog_dict$_{pi}$ where k=1
Return Assigned_Weight
No
If ∃ w | w ∈ CPT$_{pik}$
No
Is CPT$_{pi}$ empty?
- Pop tuple k - Next w ∈ CPT$_{pik}$ - k = k+1
Yes
Yes
If w ∈ CPT_catalog_dict$_{actual}$
Yes
Algorithm 3
No
Algorithm 4

Figure 20. The class weight recalculation procedure. Algorithm 10 and 11 are calculating new weights based on the observed circumstances.

The Primary CPT prediction results are reported with respect to different filtering and subseting of the dataset to reflect the performance of the presented model from important perspectives. These settings are introduced in Table 8 and we will discuss more about these in the results section.

Setting Label	**Description**
C	Complete data set
F_1	Overlook small CPT difference $(CPT_i - CPT_j \leq 10 \Rightarrow y_{0,1} = 1$
F_2	Eliminate rare CPT occurrences $(F_{CPT_i} < 4)$
F_1F_2	F_1 & F_2 $(F_1 \cap F_2)$
F_3	Recalculate CPT weight
$F_3CPT_pRVU_{max}F_1$	F_3 & (1_{st}) max weight CPT & compare with max RVU label & F_1
$F_3CPT_pRVU_{max}F_2$	F_3 & (1_{st}) max weight CPT & compare with max RVU label & F_2
$F_3CPT_pRVU_{max}F_1F_2$	F_3 & (1_{st}) max weight CPT & compare with max RVU label & F_1 & F_2
$F_3CPT_{p,s}RVU_{max}F_1$	F_3 & 1_{st} and 2_{nd} max weight CPTs & compare with max RVU label & F_1
$F_3CPT_{p,s}RVU_{max}F_2$	F_3 & 1_{st} and 2_{nd} max weight CPTs & compare with max RVU label & F_2
$F_3CPT_{p,s}RVU_{max}F_1F_2$	F_3 & 1_{st} and 2_{nd} max weight CPTs & compare with max RVU label & F_1 & F_2
$F_3CPT_p[CPT]F_1$	F_3 & (1_{st}) max weight CPT & compare with CPT list & F_1
$F_3CPT_p[CPT]F_2$	F_3 & (1_{st}) max weight CPT & compare with CPT list & F_2
$F_3CPT_p[CPT]F_1F_2$	F_3 & (1_{st}) max weight CPT & compare with CPT list & F_1 & F_2
$F_3CPT_{p,s}[CPT]F_1$	F_3 & 1_{st} and 2_{nd} max weight CPTs & compare with CPT list & F_1
$F_3CPT_{p,s}[CPT]F_2$	F_3 & 1_{st} and 2_{nd} max weight CPTs & compare with CPT list & F_2
$F_3CPT_{p,s}[CPT]F_1F_2$	F_3 & 1_{st} and 2_{nd} max weight CPTs & compare with CPT list & F_1 & F_2

Table 8. Different settings (data filtering methods) for deep result analysis

4.2 Primary CPT Prediction with respect to Surgery Duration Loss

Tree-based ML models provide an alternative to additive and linear logistic models for solving classification problems. Specifically, Random Forest technique is useful for CPT classification problem where we have a set of classification variables and a single-response class; CPT codes. Statistical inference for such models is in its infancy as a a specific feature selection underlies tree-based. In this section, our proposed method is using a collection of functions (CPT classification and surgery duration loss functions) instead of a single objective function in RF modeling. This will form a basis for building and assessing the new CPT class assignments. Then, we applied a metaheuristic optimization method, Genetic Algorithm (GA), to help in selecting the best set of final decision trees given the appropriate goal function. This section is our attempt to show the importance of custom loss function in RF model to produce more reliable durations in this effort.

- Cost Function

  Like any other ML algorithms, RF model is trained to minimize a cost function on the surgical training data. In literature, there are few well-developed cost functions that are using the capability of the available built-in libraries. In the real-world problems, such off-the-shelf loss functions are usually not well-tuned to achieve specific goals, in this case the CPT classification and surgery duration estimation problems. Hence in this study, we assigned a hyperparameter λ_i to two players of the cost function. This controls the accuracy trade-off (CPT classification vs. duration estimation) by adjusting the weight of the penalty term.

  Given the goal of predicting CPT codes with respect to minimized duration loss, we optimize the general cost function $(f(g(x, \lambda_1), h(x, \lambda_2))$ where the optimal decision depends upon magnitude of weight we want to give to both functions, CPT classification loss function g and duration loss function h which are λ_1 and λ_2. Consequently, this enables the user to choose to either predict more accurate CPT code or produce more precise surgery duration distributions based on the scheduling needs. This is formally called a Neyman-Pearson criterion [13].

  In a standard multi-label RF classification application Gini Impurity metric serves as the decision criteria in the form of objective function to determine the best splits. Suppose C is the total number of unique classes in the training dataset and $P(j)$ is the probability of randomly choosing a data point with class j, then Gini Impurity is computed as follows:

$$G = \sum_{j=1}^{C} p(j) * (p(j) - 1) \tag{7}$$

  When training a Random Forest model, the best split (in each tree in the forest) is selected by maximizing the Gini index which is the weighted Gini impurities of the branches subtracted from the original impurity. Given that the Gini gain is cable of fulfilling only the binary or multi class classification goal, it's not tuned to account for a continuous loss, e.g. duration estimation loss.

- Genetic Algorithm

  Genetic algorithm is designed to search according to the natural selection and genetics pattern mechanisms. In terms of its application in real world problems, this algorithm is mostly used when searching in non-linear and large spaces. Consequently, due to existence of many possibilities in search area, the subject-matter expert knowledge or the state-of-the-art are lacking. The idea and basis of such mechanism was first introduced by John Holland [33] in 1988. This algorithm employs principles as basis for the search such as chromosomes. Chromosomes are considered as a population of individual strings while each of them represents one possible solution to the defined problem.

In the GA context, parents are utilized to generate the generations of candidate individuals in each iteration. The children produced by the parent pairs are then used in recombination step where a crossover operator is present. Crossover includes selection of a random split point on the individual genes and generating a child individual with the genes up to the split point from the first individual parent and from the split point to the end of the individual taken from the second parent. Similarly, the second child is also inverted.

Meanwhile, a fitness value is assigned to each chromosome based on the outcome of the fitness function. This allows the algorithm to provide more chance of reproduction to highly fit chromosomes and the offspring share features acquired from their parents. This approach is different from other optimization and search methods in literature based on the following reasons [33]. This algorithm:

1. employs a function of the parameter set (resembles coding region in human DNA), not the parameters themselves.
2. uses payoff information, not derivatives or other auxiliary knowledge. The value of an objective function feeds back to direct a search.
3. utilizes probabilistic transition rules (not deterministic ones). Then, the sub-optimal fitness may be achieved while the best fitness value is not always guaranteed.
4. initiates the search process using a population of points, not from just a single point and is carried out from generation to generation until convergence is reached.

GA generally consists of three operators: 1. selection, 2. recombination and, 3. evaluation. Assume $P(g_t)$ is the population of individuals (set of X_is) at generation t, a simple GA will have the following structure:

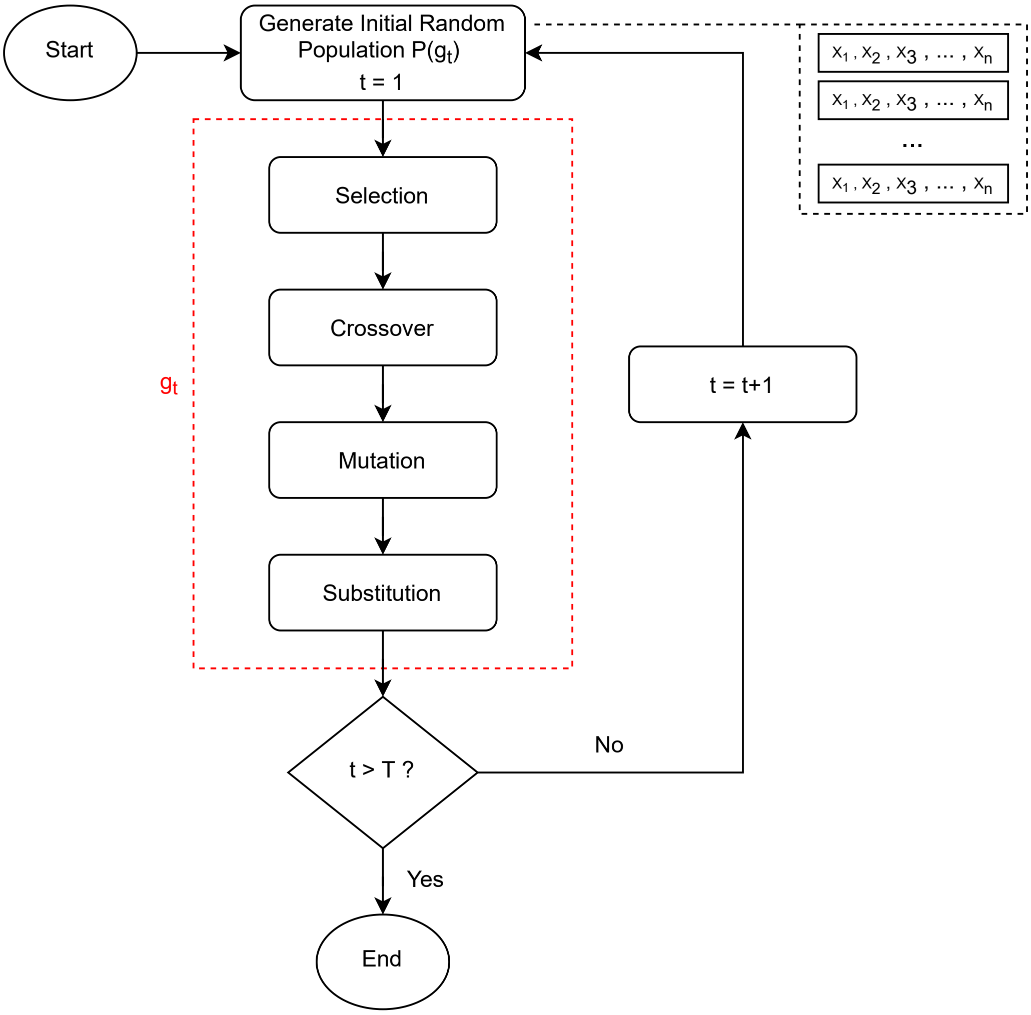

Start
Generate Initial Random Population P(g_t) t = 1
$x_1, x_2, x_3, \ldots, x_n$
$x_1, x_2, x_3, \ldots, x_n$
...
$x_1, x_2, x_3, \ldots, x_n$
Selection
Crossover
g_t
Mutation
Substitution
t = t+1
t > T ?
No
Yes
End

Figure 21. A simple GA structure.

The fitness function evaluates the individuals (set of X_is) in each population in every iteration ($t < T$). In the selection process, individuals are selected and their genes are passed to the next generation based on their fitness function value. The traditional Genetic algorithm employs a selection method where the selection probabilities are proportional to their corresponding fitness values. This selection method (so-called roulette wheel selection) reassures that highly fit individuals have a higher probability of being selected for the next step. The next step is recombination process which includes two operators; crossover and mutation. The crossover operator randomly chooses a crossover cite, cuts individuals into two sub-strings, and swaps the tail sub-strings to generate two offspring individuals.

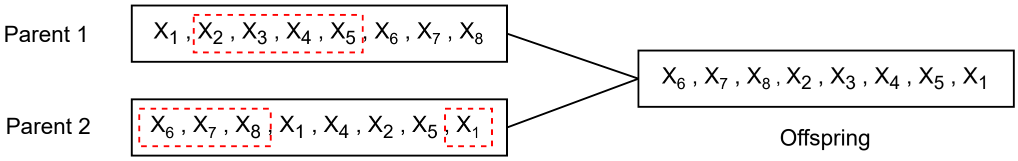

Parent 1
X_1, X_2, X_3, X_4, X_5, X_6, X_7, X_8
Parent 2
X_6, X_7, X_8, X_1, X_4, X_2, X_5, X_1
X_6, X_7, X_8, X_2, X_3, X_4, X_5, X_1
Offspring

Figure 22. GA crossover example.

Mutation process modifies the value of random genes in each individual to help the algorithm prevent getting stuck on a local optimal or in other words premature convergence.

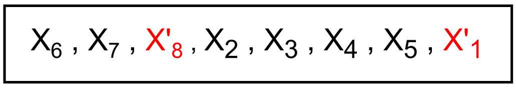


Figure 23. GA mutation example.

- GA application in RF loss function modification
  - Revised Loss Function

    The revised loss function aims to select the top k trees based on their CPT prediction accuracy as well as the median duration error (of the predicted CPT vs the actual CPT). Therefore, first define the loss terms for these two objectives separately, and then introduce the final objective function which is used in GA algorithm as a next step.

    For each top k selected trees we first measure the CPT accuracy. Each tree predicts the CPT and it can be compare to the actual CPT of the case and calculate the score. The score is 0 if the prediction is wrong and 100 vice versa. For the duration objective, the predicted CPT label by each tree has a median actual surgery duration which can be compared to the one for the actual CPT label. Therefore, we can calculate the duration error given the median duration differences for predicted and actual CPTs. We also normalize the duration errors and subtract that from 1 so that the duration score can be put in the same scale as the CPT prediction objective. Since the higher the CPT accuracy is the better the model performed, we can describe the same behavior for $100 * (1 - NormalizedDurationError)$. Also we find the optimal importance coefficients that leads the model to produce the best results with not sacrificing CPT score but also improve the duration estimation. The sweet point for predicting CPTs when focusing on duration results is 0.4 and 0.6 for CPT score and duration score, respectively. Then, simply the sum of $0.4 *$

$CPTAccuracyScore_{(0|100)} + 0.6 * 100 * (1 - NormalizedDurationError)$ for all k trees in the subset forms the final objective of GA in selecting the best set of k trees from N total trees in RF with respect to both perspectives; CPT prediction and duration estimation.

- GA and RF bagging

  Genetic algorithm employs the revised cost function (explained in previous item) to choose the best set of k (here $k = 100$) decision trees out of N (here $N = 1000$) total trees that fits the CPT classification and surgery duration estimation goals. In other words, in each iteration, the selected tree population with the best fitness score is passed to the next steps; crossover, and mutation. Therefore, in order to properly adopt the GA method in this use-case, we define the individuals as the series of constructed trees in the process of training Random Forest algorithm and obtaining the $N = 1000$ trees in the forest. Then, the random populations (sets of trees each set with size of 100 trees) are drawn from the complete individual set. The populations can be represented as set of unique binary individuals. Each binary individual reflects the presence (gene is encoded as 1) and absence (gene is encoded as 0) of the randomly selected trees. For crossover and mutation steps, we use the traditional methods where as an example in mutation the value of a random gene is modified (e.g. from 0 to 1 or vice versa); switching on / off the trees existence in each individual of the whole population.

  Collectively, as the second step in the attempt to predict CPT code with respect to duration loss, we use GA to optimize the tree selection at the bootstrap stage of Random Forest model given the new objective function. Similarly, [60] recommend an enhanced decision tree algorithm for user classification in mobile application, which employed genetic algorithm to optimize the results of the decision tree algorithm. However, Liu et. al optimized the generated rules within decision tree splits to come up with the best set of rules in producing the desired classification outcome. In this study, we focus on the weak learners (trees) in Random Forest and improve the class selection (i.e. bagging) based on the optimized objective function. Each tree in the forest is generated through assigning a new feature in the split node and produce a class given the final class votes in the leave nodes. We also performed the grid search on the parameters of GA to find the best set size (k), mutation probability bound ($MUTPB$) and crossover probability bound ($CXPB$).

4.3 Multi-CPT Prediction

Manual CPT coding has become a huge burden in U.S. Healthcare due to surge in surgery case volume. This effort has even led to significant error in coding (human error) which cost hospital income loss according to [1]. Consequently, ML application in such coding practices has recently attracted interests from efficient healthcare management and effective decision making perspective. In this section, we develop a multi-channel deep convolutional neural network to extract information from the surgery descriptions and other categorical and continuous features through multiple layers and activation functions.

One simple neural network model for text and sentiment analysis employs single word-embedding layer in the context of one-dimensional convolutional neural network (CNN) model. A convolutional neural network consists of stacked, layered, operations. There are two types of layers, convolutional, and spacial pooling. The convolutional layers extract feature maps by applying several trainable filters to the input, before applying a nonlinear activation function (e.g. relu and sigmoid) to the result. The spacial pooling layers (with pool size equal to 2) operate in a similar fashion by applying an operation to a receptive field which is moved over the input feature map. The operation is designed to down-sample the input so the resulting feature map has reduced dimensions. The two layers are stacked alternatively, with the idea being that the complexity of the features extracted increases with the depth of the network. The kernel size in the convolutional layer defines the amount of words to consider as the convolution is passed across the input text document, providing a grouping parameter.

The central concept of the convolutional layer is the convolution operation. Let the kernel, w, be a $k \times k$ dimensional matrix. This kernel will operate on the output of the preceding layer, x. The output from the convolution can be calculated as follows:

$$w * x_{ij} = \sum_{m} \sum_{n} w_{mn} x_{i-m,j-n}$$

Where (m, n) spans the index set of the kernel which is center originated, i.e. $w_{0,0}$ is the center element of the kernel. The patch of x involved in the sum at each step is referred to as the receptive field. As the operation is repeated for every index of x, the receptive field slides across the input. The resulting output of the convolution is referred to as a feature map.

Such model has been developed with more complexity in the literature by adding multiple parallel convolutional neural networks [49]. These networks read the input data with various kernel sizes which effectively generates a multi-channel CNN for text layers that analyze text data with different n-gram word combinations. The kernel size parameter in convolutional layer represents the number of tokens to consider as the layer is passed across the input text data, in a grouping manner. Normally, a standard model for text labeling is supported by an Embedding layer as encoded text input, followed by a one dimensional convolutional pooling layer, and a prediction output layer with respect to output dimension. A multi-channel convolutional neural network for text classification integrates multiple versions of the standard model with different sizes of kernels. This capability of CNN allows processing the text information at several resolutions or multiple n-grams simultaneously, while the final model learns how to best integrate these interpretations.

- Model Architecture
  - Pre-trained Biomedical Embedding

    In any ML modeling, textual data needs to be encoded in the form of vectors. The initial step is fitting a Keras Tokenizer [3, 76] on the the cleaned surgery descriptions of the trainig dataset. We will use this tokenizer to both define the medical vocabulary for the embedding layer and encode the descriptions as numerical values. We pad all surgery description records (as sequences) to the fixed length by taking the maximum length of input sequences. For the embedding layers, we also need to extract the size of the vocabulary. In this thesis, we employ pre-trained biomedical embeddings,: 1. BioNLP with 200-dimensional word vectors trained on PubMed with 2.89B tokens using word2vec embedding method, and 2. FastText with 200-dimensional word embeddings trained on PubMed and MIMIC-III.

    Word2vec is a well-established word embedding method, referred as a predictive model learns word embeddings by predicting the words based on their context [64]. Two main techniques exist for word2vec; CBOW and skip gram. Continuous bag of words (known as CBOW) calculates the word conditional probability as target given the combination of words around it while, skip gram predicts the surrounding context words by setting the target. Both approaches can be conceptualised as a shallow NN. In skip-gram method, the target word is entered as the input layer while the surrounding context tokens are in the output layer. Generally, both the target word and it's relative context are employed in encoding a vector representation of the word.

FastText is another word embedding method that has been used in this research [11]. It is a Word2vec extension where words are represented by a set of n-grams characters within them. Likewise, it has both skip gram and CBOW models. The benefit of using character n-grams of words is that it allows the embeddings to learn vectors in morphological tasks as well as extracting information in case of unstructured text. With respect to unstructured text, it allows to learn the words that may not exist in the embeddings (e.g. typos) while it's similar form exists. Such n-gram embeddings can be utilized to compute a vector corresponding to the n-grams of an out-of-dictionary token, essentially due to its built-in mechanism of character n-grams.

– Define Model

Such deep NN structure was first introduced by [49]. Kim demonstrated static and dynamic embeddings, however this research simplifies the method and focuses on the importance of various kernel sizes in the proposed embedding layers. This section explains architecture of our multi-channel NN model which is demonstrated in Figure 18. This network has three input channels with different n-grams, such as 2-grams, 3-grams, and 6-grams in word embeddings of various sources (PudMed, and FastText) for processing text information. We also incorporated additional categorical and continuous features as input layers. Each channel consists of these components:

- ∗ Input layers for the surgery description text data (captures text sequence lengths);
- ∗ Embedding layers to encode the text using embedding methods and sources discussed above (captures the vocabulary size and vector of k-dimensional word

representations; here 200);

- Input layers for other categorical and continuous features (age, case type, scheduled duration, and etc.);
- Single-dimensional convolution layer with 32 filters, and kernel size as the number of words to pass simultaneously;
- Dropout layers to prevent NN model from overfitting. We perform grid search to discover the best dropout probability for this model and our dataset, as well as how sensitive it is to the dropout rate since the more sensitive a model is, the more unstable results we may expect which implies that it very well can benefit from increasing the data size;
- Max-pooling layers for consolidating the output from convolution layers;
- Flatten layer to reduce the multi-dimensional output layer to support the number of dimensions defined with respect to multi-CPT prediction work stream;
- Concatenate layers for concatenating the text and other features in each channel, and merging three channels and use it in the output layer;
- Dense layers to process output with respect to the output shape, the output layers from 3 channels are merged into one final dense layer;

As a result of running the presented multi-channel CNN, the summery of the model architecture and network are presented in Figure 24, and Table 9.

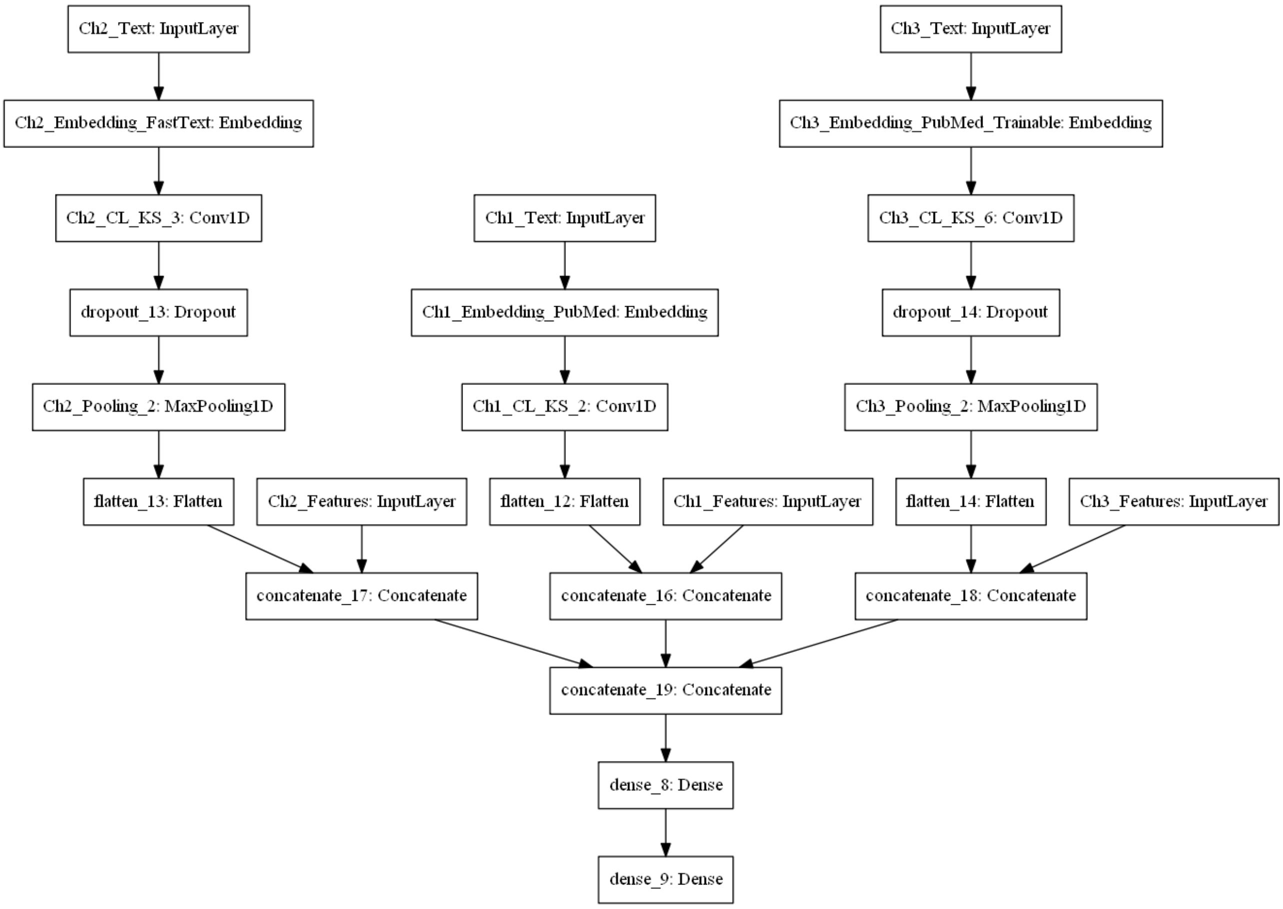
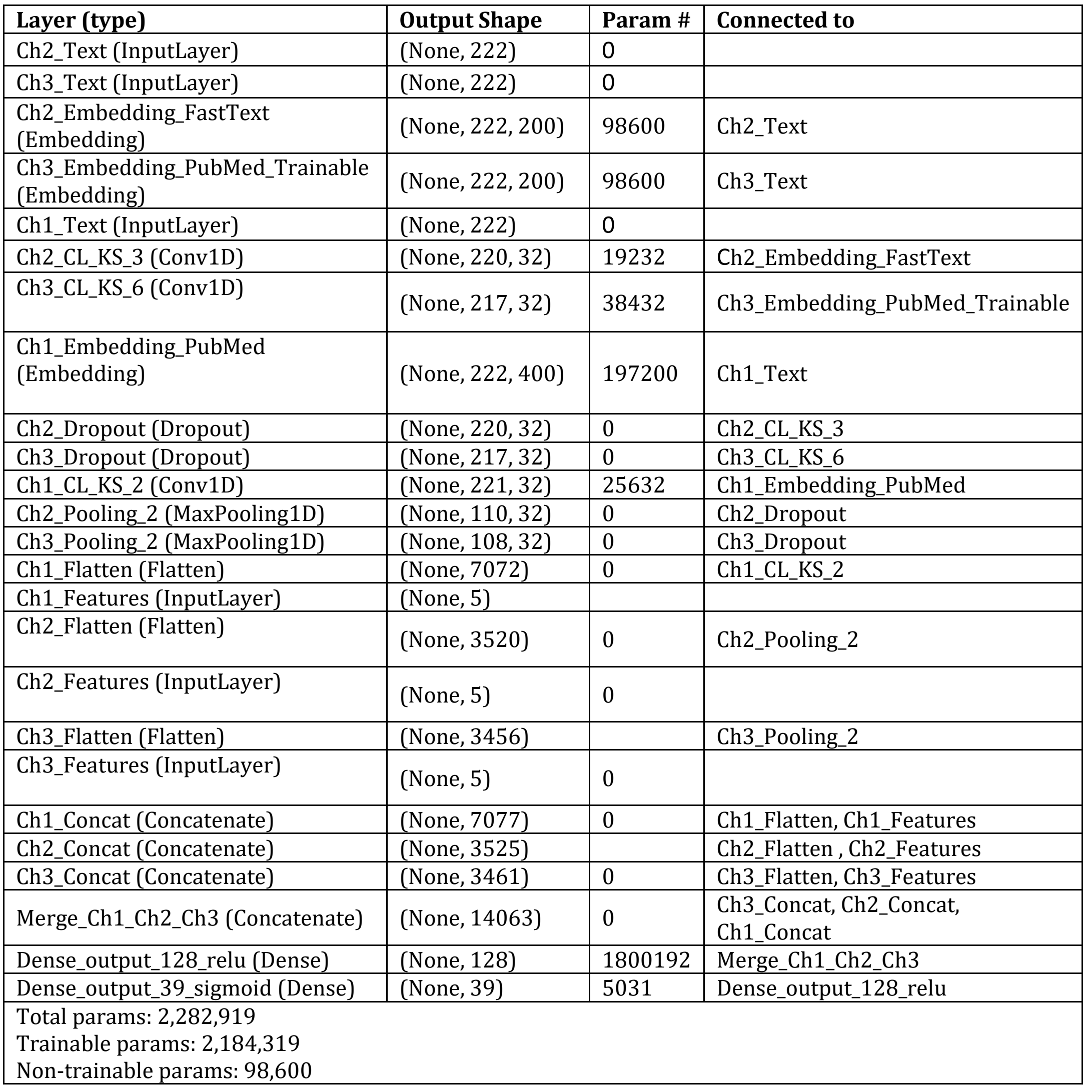


Figure 24. Multi-channel CNN model architecture.

As indicated above, we have defined 3 embedding layers for each convolution channel. First embedding layer uses the tensorflow pre-trained embedding [2], second embedding layer embedding employs weights from a pre-trained PubMed word2vec embedding matrix with holding the embedding matrix constant during training (trainable parameter is set to False) or technically freezing the layer, and the last layer utilizes Fasttext embedding matrix while it keeps changing the trainable weights during fit training or in the custom loop which depends on embedding weights to apply gradient updates.



Layer (type)	Output Shape	Param #	Connected to
Ch2_Text (InputLayer)	(None, 222)	0	
Ch3_Text (InputLayer)	(None, 222)	0	
Ch2_Embedding_FastText (Embedding)	(None, 222, 200)	98600	Ch2_Text
Ch3_Embedding_PubMed_Trainable (Embedding)	(None, 222, 200)	98600	Ch3_Text
Ch1_Text (InputLayer)	(None, 222)	0	
Ch2_CL_KS_3 (Conv1D)	(None, 220, 32)	19232	Ch2_Embedding_FastText
Ch3_CL_KS_6 (Conv1D)	(None, 217, 32)	38432	Ch3_Embedding_PubMed_Trainable
Ch1_Embedding_PubMed (Embedding)	(None, 222, 400)	197200	Ch1_Text
Ch2_Dropout (Dropout)	(None, 220, 32)	0	Ch2_CL_KS_3
Ch3_Dropout (Dropout)	(None, 217, 32)	0	Ch3_CL_KS_6
Ch1_CL_KS_2 (Conv1D)	(None, 221, 32)	25632	Ch1_Embedding_PubMed
Ch2_Pooling_2 (MaxPooling1D)	(None, 110, 32)	0	Ch2_Dropout
Ch3_Pooling_2 (MaxPooling1D)	(None, 108, 32)	0	Ch3_Dropout
Ch1_Flatten (Flatten)	(None, 7072)	0	Ch1_CL_KS_2
Ch1_Features (InputLayer)	(None, 5)		
Ch2_Flatten (Flatten)	(None, 3520)	0	Ch2_Pooling_2
Ch2_Features (InputLayer)	(None, 5)	0	
Ch3_Flatten (Flatten)	(None, 3456)		Ch3_Pooling_2
Ch3_Features (InputLayer)	(None, 5)	0	
Ch1_Concat (Concatenate)	(None, 7077)	0	Ch1_Flatten, Ch1_Features
Ch2_Concat (Concatenate)	(None, 3525)		Ch2_Flatten , Ch2_Features
Ch3_Concat (Concatenate)	(None, 3461)	0	Ch3_Flatten, Ch3_Features
Merge_Ch1_Ch2_Ch3 (Concatenate)	(None, 14063)	0	Ch3_Concat, Ch2_Concat, Ch1_Concat
Dense_output_128_relu (Dense)	(None, 128)	1800192	Merge_Ch1_Ch2_Ch3
Dense_output_39_sigmoid (Dense)	(None, 39)	5031	Dense_output_128_relu
Total params: 2,282,919 Trainable params: 2,184,319 Non-trainable params: 98,600			

Table 9. Multi-channel CNN model summary.

4.4 Evaluation Metric

An important step towards building a model is defining how we measure its performance. Implicitly, this is done through the construction of a *Loss function.* The models we examine in single CPT prediction section do not employ novel loss functions, so delving into their construction is not warranted. The metrics used when measuring the performance of classification models (with multi-label, single-class output), specifically, predicion, recall, and accuracy are however of interest.

- Precision, Recall, & Accuracy

  The precision and recall of the CPT prediction model refers to its ability to correctly label the surgery cases within each specialty data. The accuracy is the ratio of corrected predicted CPTs over the total number of data points. Before we can define these measurements in multi-label single class classification problem, we must first introduce the following quantities:

True Positive (TP): Number of primary CPTs correctly labelled.

False Positive (FP): Number of incorrect primary PT predictions.

True Negative (TN): Correct non-prediction, not usually relevant.

False Negative (FN): Number of objects missed by model.

Precision measures the accuracy of the model, i.e. how many of the predictions are correct while recall is how many of the surgical cases the model correctly labels. These two metrics are the basis for how both text mining (as described in the first section) and classification (binary or multi label) problems are evaluated. They are computed by substituting the above quantities in formulas (7), (9), and (8).

$$Precision = \frac{TP}{TP + FP} \tag{8}$$

$$Recall = \frac{TP}{TP + FN} \tag{9}$$

$$Accuracy = \frac{TP}{TP + FP + FN} \tag{10}$$

In the case of multi-CPT prediction as a multi-label classification problem, we first calculate the subset accuracy. The subset accuracy is 1 if the full set of predicted CPTs for a surgery case matches with the true set of CPT labels (primary and secondary CPTs). In addition, we assigned accuracy 0.5 if one of the predicted CPTs, either primary or secondary) exist in the true CPT label set.

4.5 CPT Prediction Results

4.5.1 CPT and Duration Prediction Results

As stated in the previous chapter, the dataset consists of both categorical and continuous features with nearly 10000 data rows. The CPT codes describe the details of procedure and is considered as target in this step. The surgery durations (either scheduled or actual) varies based on the procedure tasks summarized as CPTs. Figure 25 shows the distribution of 20 top frequent CPT codes (noted as significant labels) within each specialty dataset. The average scheduled durations of these CPT codes are also demonstrated in the heatmap plot. For instance, in Cardio dataset CPTs "76376" and "93320" are surgery codes with highest average scheduled durations.

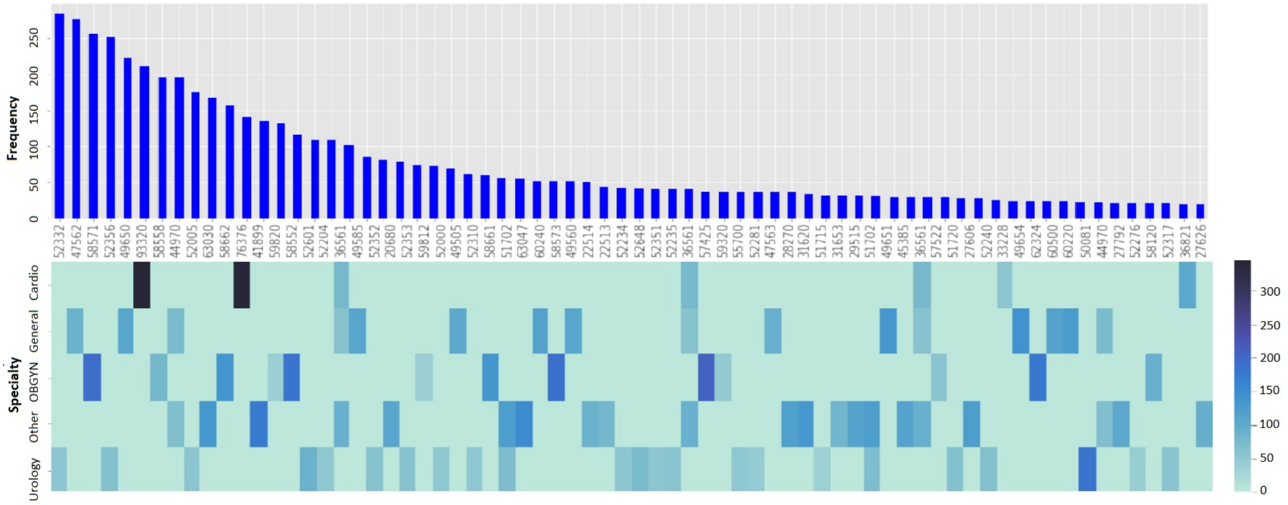


Figure 25. Most frequent CPTs distribution and average durations per specialty.

The general framework of this research includes the following phases: CPT classification and duration estimation. In the interest of performance evaluation, the classification models are evaluated by the accuracy score which is a popular metric for reporting the performance rate. We also report the models' performance by precision and recall.

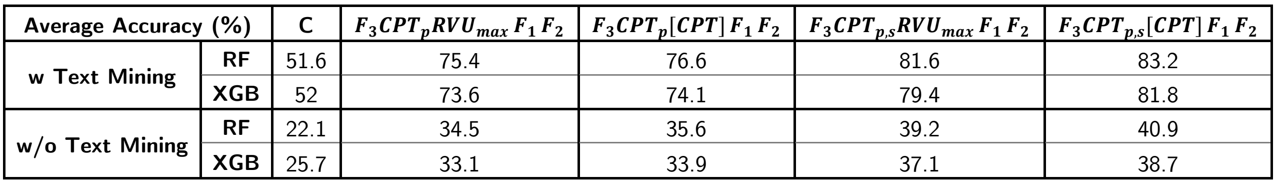
4.5.2 Primary CPT prediction

We have tested the performance of several multi-class classification methods for each specialty. Random forest modeling approach has been observed as the superior method and thus has been has been fitted on each specialty data; the performance is reported in Table. 10. The specialty "Other" can be decomposed into more specific specialty types (e.g. ENT, Orthopedics, and so on) as more future observations of the same discipline are added to the data set. We also compare the performance of original RF model and RF + CWR extension (our method) with the state-of-the-art Neural Network model (see Table. 10).

Algorithm:	Random Forest																Complementary Weight Recalcultion																Neural Net Model															
Compare to:	CPT_RVUmax								CPT_set								CPT_RVUmax								CPT_set								CPT_RVUmax								CPT_set							
Compare with predicted:	1 CPT				2 CPTs				1 CPT				2 CPTs				1 CPT				2 CPTs				1 CPT				2 CPTs				1 CPT				2 CPTs				1 CPT				2 CPTs			
Filters:	C	F1	F2	F1F2	C	F1	F2	F1F2	C	F1	F2	F1F2	C	F1	F2	F1F2	C	F1	F2	F1F2	C	F1	F2	F1F2	C	F1	F2	F1F2	C	F1	F2	F1F2	C	F1	F2	F1F2	C	F1	F2	F1F2	C	F1	F2	F1F2	C	F1	F2	F1F2
Cardio	45	53	57	62	49	59	58	71	50	54	60	67	54	62	60	72	54	61	68	72	68	76	86	89	55	62	69	73	69	75	86	88	39	45	48	54	39	46	49	54	39	46	50	56	39	47	51	57
General	68	78	77	87	71	73	76	81	69	78	79	88	71	73	76	81	70	79	79	88	74	81	84	90	71	79	80	88	76	83	84	91	15	28	16	30	15	28	17	30	16	29	18	31	16	30	19	33
Urology	52	62	56	71	55	67	63	72	54	63	56	71	57	69	65	74	57	66	62	76	60	74	62	78	58	67	71	78	63	74	74	79	20	24	22	26	20	24	22	26	22	24	23	28	23	26	23	28
OBGYN	57	74	64	81	59	74	65	82	58	74	67	82	61	74	68	84	63	77	70	83	68	80	76	84	66	79	72	85	75	83	82	90	34	47	37	50	34	48	38	51	34	47	39	53	36	47	39	55
Other	36	51	54	68	37	53	56	69	38	52	56	69	40	54	58	70	39	52	55	65	40	53	55	66	44	57	58	71	44	58	59	71	17	20	22	25	17	20	23	26	17	21	23	27	17	22	25	27
Average	52	64	62	74	54	65	64	75	54	64	64	75	57	66	65	76	57	67	67	77	62	73	73	81	59	69	70	79	65	75	77	84	25	33	29	37	25	33	30	37	26	33	31	39	26	34	31	40

Table 10. CPT prediction accuracy measures under different filter combinations and accuracy calculation approaches for each specialty.

The variations of analyzing the accuracy are defined in Table 4. These settings are originally generated from predominant elements: data filtering approaches as F_1, and F_2, or the performance improvement settings as $F3$ (provides either 1_{st} or $[1_{st}, 2_{nd}]$ highest weight CPT) or the comparison baselines as CPT of RVU_{max}, and [CPT] (provides all possible CPT codes in CPT list). These filters are defined to better understand the nature of data and prediction schemes with respect to the weighting approach. Each closed break line in radar graph in Fig. 20 is showing the accuracy variations of each specialty data under different filters and settings. Moreover, the text mining approach improves the CPT prediction performance and hence the surgery durations significantly (discussed in the next chapter). The average accuracy of CPT prediction models, XGB and RF, are represented in Table 11 with and without the transformed text features in the feature set.



Average Accuracy (%)		C	$F_3CPT_pRVU_{max}\,F_1\,F_2$	$F_3CPT_p[CPT]\,F_1\,F_2$	$F_3CPT_{p,s}RVU_{max}\,F_1\,F_2$	$F_3CPT_{p,s}[CPT]\,F_1\,F_2$
w Text Mining	RF	51.6	75.4	76.6	81.6	83.2
	XGB	52	73.6	74.1	79.4	81.8
w/o Text Mining	RF	22.1	34.5	35.6	39.2	40.9
	XGB	25.7	33.1	33.9	37.1	38.7

Table 11. Plot of average CPT prediction accuracy under optimal settings for XGB and RF models.

To better demonstrate the performance of our prediction model, we compute the weighted average precision and recall. Knowing that the CPT dataset is unbalanced and includes at least 800 unique CPTs, precision and recall metrics are weighted in each specialty dataset. Therefore, we calculate the overall precision and recall given the specialty-specific performance measures and weight them based on the sizes of each dataset. The total weighted average recall and precision for the Neural Net model (as state-of-the-art model) are 0.22 and 0.23 while for the presented model (CWR) are 0.52 and 0.45, respectively. These results are reported based on the CPT predictions which are drawn from complete data (C), and the model predicts the primary CPT (accuracy@2). The true label is chosen based on maximum RVU score. Given that F2 is defined as one of the substantial filters, we report the precision and recall of CWR and NN while this filtering method is present. With respect to F2 filter, the recall and precision of NN are 0.26 and 0.28 and for CWR are 0.64 and 0.62 which are close to their corresponding accuracy scores depicted in Table 11. While the highest accuracy (84%) is reported as accuracy@2 when compared with CPTset and presence of F1 and F2 methods, the recall and precision are also calculated as 0.86, and 0.85, respectively.

4.5.3 Multi-CPT Prediction

We ran the deep multi-channel model with the full CPT dataset while filtering out those cases with CPT frequencies less than 8. After filtering the rare CPT cases, we are left with 738 surgery cases with 2 CPT labels and 6946 surgery cases with single CPT label. For each surgery in multi CPT data, we consider 0.8 reward if the primary CPT, and 0.2 if the secondary CPT are predicted accurately. Hence, for each case if both primary and secondary CPTs are correctly predicted then the score will be 1. The accuracy, weighted precision, and recall (based on the CPT counts in the dataset) of multi CPT data are 0.7, 0.69, and 0.61, respectively. The accuracy breakdown for each specialty, namely Cardio, General, OBGYN, Urology, and Other is as follows: 0.77, 0.75, 0.64, 0.72, and 0.54. We also predicted for single CPT cases using the same multi-channel structure and the accuracy, weighted precision, and recall are 0.45, 0.41, and 0.36. The multi-channel model performs worse than the presented CWR model for primary CPTs, however it performs better than the NN model in previous section. We think that if more multi CPT surgery cases are added to the dataset (with frequent CPT occurrences), the multi-channel model can perform better since such models have many parameters to learn which highlights the need for a well-representative dataset.

5 Surgery Duration Prediction

The ultimate goal of this chapter is predicting / estimating the surgery procedure durations. The duration prediction analysis is performed in two ways: directly from input data using a regression model and indirectly from the classified CPT codes using statistical sampling. In indirect method, we calculate the distribution characteristics of surgery durations given the actual durations of surgery cases for each CPT code. The CPT codes are produced by the presented methods for primary CPT prediction. Direct and indirect approaches are described in greater detail below. We further evaluate the performance of these models in the "results" chapter.

5.1 Direct Method

The first approach focuses on prediction of the point-estimate surgery durations given the set of correlated features by state-of-the-art regression models such as Decision Tree Regression (DTR), Multiple Linear Regression (MLR), Random Forest (RF), Support Vector Regression (SVR), and Multilayer Perceptron (MLP). The feature set employed in these models include the numerical transition of text feature (surgery descriptions), surgeon, specialty, case type, and patient age. The purpose of running ML models is to create a comparable baseline which we can report the current state accuracy in duration prediction if the commercial models are to be used.We found that the aforementioned models are widely used in literature to predict continuous variable such as duration, or in similar efforts [92]. Below we briefly explain the mechanism of each machine learning model.

- Decision Tree Regression (DTR) & Random Forest Regression (RFR)

  Decision tree models are known as the most popular machine learning methods [19] in many classification applications due to their simplicity and intelligibility. While a wide range of such methods application can be found under classification problems, DTR-like analysis is when the predicted target (y) is a continuous variable (y $\in R$). One of the key elements in decision tree regression models is variance reduction, VR, of a split decision at node N in each tree due to existence of continuous variable as the target. It aims to reduce the total variance of the continuous target variable y as a result of the split at node N.

  Random Forest regression model (known as ensemble method) construct more than one decision tree to make the predictions by re-sampling training data with replacement, repeatedly. The nonlinear nature of RF model outperforms other linear algorithms in many applications. However, it is also important to know the data structure and the major limitation of this model; that Random Forest can't extrapolate. It predicts the continuous target within the average of the labels in the training data. In this sense, RF algorithm is quite similar to K-Nearest Neighbor algorithm. In other words, the range of predictions that the Random Forest model produce is bound by the highest and lowest continuous target in the data. This behavior becomes problematic when the training inputs and target labels differ way greater than their distribution metrics (e.g. outside of variance limits from the average). This issue, so-called co-variate shift, is a limitation in most of the machine learning models especially in Random Forest, due to lack of extrapolation.

- Multiple Linear Regression (MLR)

  The goal of multiple linear regression models (MLR) is to find the closest expected function ($\hat{y}$) that best explains the linear relationship between the independent variables (X_i) and response target given the training dataset. Due to having more than a single predictor ($i > 1$), we applied MLR which is an extension of ordinary least-squares regression. Generally, the assumptions of this model are homoscedasticity and independence of independent variables.

- Support Vector Regression (SVR)

  Support Vector Regression model provides the flexibility to define the extend of the residual that should be considered as the model error and find the fitted line or hyperplane accordingly. Despite the MLR regression, SVR objective function is to minimize the coefficients or l2-norm of the coefficient vector instead of squared error. Ultimately, the SVR error (ϵ) is controlled through the model's constraints. In the constraints, we can choose the desired lower limit (margin) for the absolute error (so-called the maximum error). The model can be tuned using the margin values as the most important hyper parameter to achieve the best accuracy.

- Multilayer Perceptron (MLP)

  MLP, as one important form of Neural Network models, optimizes the squared loss using iterative methods as SGD, LBFGS, or ADAM [50]. MLP is trained using back-propagation [80] method; an extension of gradient descent in which the gradients are computed using back-propagation with no activation function (equivalent to identity function) in the final output layer, hence the loss function is simply squared error. Also, a regularization term can be added to the cost function which may shrink parameters in the model to avoid over-fitting.

While each method predicts the point estimate duration as continuous variable for each surgery case, we need to make sure that the reported durations are in a form of distribution. This is because the output of indirect method (described in the next subsection) is the duration distribution parameters (mean and variance) for the predicted CPTs of surgical cases. Therefore, the distribution characteristics of point estimate duration predictions are calculated by simply sampling based on the actual CPT code assuming that the correct CPT code is known which presents a best case scenario for predicting duration distributions. This assumption also highlights the importance of CPT study as the CPT codes are eventually unknown prior to surgeries.

5.2 Indirect Method

The indirect approach describes the procedure of predicting surgery durations with respect to an intermediate step; the CPT prediction method (described in previous step). Consequently, given the predicted CPT codes we can obtain the duration distribution parameters such as mean and variance for each data record in surgery schedule. Given that the study predicts primary CPT is the first place and then predict the secondary CPT, we can compute the durations using both steps' output, single and multiple CPT predictions, and compare against the direct method. Moreover, we calculate duration distribution specifications using the revised RF classification with respect to the new cost function as another alternative in our comparison scheme. The duration estimation alternatives utilized in our comparison engine are demonstrated in the table below (Table 12s).

Method	Model	Assumption	Input	Output
Indirect	RF + CWR	Cleaned surgical descriptions transformed to vector (TFIDF)	Initial Feature Set	Primary CPT Code $f^d_{CWR}(\mu_1, V_1)$
	RF + Revised Loss	Cleaned surgical descriptions transformed to vector (TFIDF)	Initial Feature Set	Primary CPT Code $f^d_{RF_L}(\mu_3, V_3)$
Direct	DTR	Cleaned surgical descriptions transformed to vector (TFIDF)	Initial Feature Set	$f^d_{DTR}(\mu_4, V_4)$
	RFR		Initial Feature Set	$f^d_{RFR}(\mu_5, V_5)$
	MLR		Initial Feature Set	$f^d_{MLR}(\mu_6, V_6)$
	SVR	Actual CPT code is known	Initial Feature Set	$f^d_{SVR}(\mu_7, V_7)$

Table 12. Actual duration distribution estimation using different proposed alternatives.

In Table 12, the initial feature set (described in the Classification section) includes surgeon ID, patient's age, case type, and text information. Also, the assumption column describes the method used for employing text information in each model in addition to the assumption of actual CPT existence in the distribution estimation step. Moreover, the output column represents the outcome of each model; actual surgery duration distribution paired with CPT code(s) (for the models in the indirect method).

5.3 Evaluation Metrics

Mean squared error (MSE), mean absolute error (MAE), duration standard error (SE), and adjusted R-squared or coefficient of determination are computed to evaluate and compare the regression model performances. The standard error of duration estimations represents the standard deviation of the absolute difference between actual duration and predicted values. The Mean absolute error represents the average of the absolute difference between the actual and predicted values in the dataset. It measures the average of the residuals in the dataset. The adjusted R-squared is used to compare the goodness-of-fit for models which include various numbers of independent variables. The reason is that R-squared increases when more variables are added to the regression model. As a matter of fact, R-squared never declines even if there is a chance correlation among the added independent variables. A model which includes more variables than another regression model may appear as it fits better solely due to existence of more independent variables. Mathematically, R-squared is computed as following while y is the actual duration, $\hat{y}$ is the predicted duration, and $\bar{y}$ is the average duration for i observations in range of {1,...,N}:

$$R^2 = 1 - \frac{\sum_{i=1}^{N}(y_i - \hat{y}_i)}{\sum_{i=1}^{N}(y_i - \bar{y}_i)} = 1 - \frac{SS_e}{SS_T} \tag{11}$$

The adjusted R-squared measures the proportion of variation explained by only those highly correlated independent variables which contribute in explaining the target variable and it penalizes the performance for including independent variable that are loosely contributes to predicting the dependent variable. We calculate this measure in our analysis due to having different number of features in the specialty dataset. The adjusted R-squared formula is:

$$\bar{R}^2 = 1 - \frac{\frac{SS_e}{df_e}}{\frac{SS_T}{df_T}} \tag{12}$$

In equation (12), df_T is the total degrees of freedom or $n-1$ in the estimate of population variance of the target (y), and df_e is the error degrees of freedom or $n-p-1$ in the estimate of underlying population error variance. Then, adjusted R-squared is computed given the R-squared, number of predictors (p), and total sample size (N):

$$\bar{R}^2 = 1 - \frac{(1-R^2)(N-1)}{N-p-1} \tag{13}$$

5.4 Duration Estimation Results

We have compared the performance of our duration prediction models (indirect methods) against the state-of-the-art models (direct methods). The indirect approach includes two methods ; 1. two-step CPT classification (RF and CWR), and 2. CPT classification using the revised loss function in RF algorithm. After predicting the surgery CPT code, we estimate the surgery case duration distribution using empirical data. The duration errors are analyzed for each CPT using performance metrics such as adjusted R-squared (equation 12), mean squared error or MSE, standard error or SE, and mean absolute error or MAE. Moreover, 2-sample Kolmogorov-Smirnov test [63] is deployed for assessing the duration distributions (output of direct method for duration prediction). In Tables 13, the performance metrics are presented based on two perspectives; 1. CPT known, and 2. CPT unknown. Below we include more explanation of what the values in this table represent under these two cases.

The results set in the first column set of Table 13, "CPT known" is considered as an assumption for replicating the reality of processes in OR department of a hospital where the actual CPT code is unknown prior to the surgery is performed. Hence, the duration estimation can vary when the CPT is unknown. The reason is that the CPT codes highly tie to the procedure details and standard durations from historical pattern. Then, if the CPT codes are known for surgery cases pre-surgery, the surgery duration distribution and its key characteristics such as mean, median, and standard deviation can be used to guide the scheduler in correctly initiating the estimation of surgery durations. Given the distribution of actual durations based on CPT codes, the scheduler can provide more accurate duration estimations. This entitles the necessity of predicting the CPT (CPT classification study contribution) and estimating the duration distribution based on the classification results.

For calculating the error estimates shown in Table 13, we use two approaches described in equations (14)-(16). In these equations, we use different estimated surgery durations to show the performance we can obtain with employing either the hospital system (equation 14 which indicates the current state) or traditional machine learning models [18, 44, 58, 70] (equation 15 when CPT is unknown). In equation 15, the errors of state-of-the-art models (SVR, DTR, RFR, and MLP) are calculated based on actual duration data points ($ActualDuration_i$ which is also indicated as the target variables (see table 5)) and the predicted actual durations

$(Pred(ActualDuration_i))$ for each surgery case (i). On the other hand, we calculate some performance measures by bringing the "CPT known" assumption for the same baseline models to introduce an upper bound or a perfect scenario where accurate CPT codes are always known pre-surgery. These upper bound performances are computed based on the error terms from equation (16) for state-of-the-art models and equation (17) for hospital system.

$$e_{p_i,p_i} = |ScheduledDuration_i - ActualDuration_i| \tag{14}$$

$$e_{p_i,p_i} = |ActualDuration_i - Pred(ActualDuration_i)| \tag{15}$$

$$e_{d_i,d_i} = |Avg_{ActualCPT(ActualDuration)} - Avg_{PredCPT(ActualDuration)}| \tag{16}$$

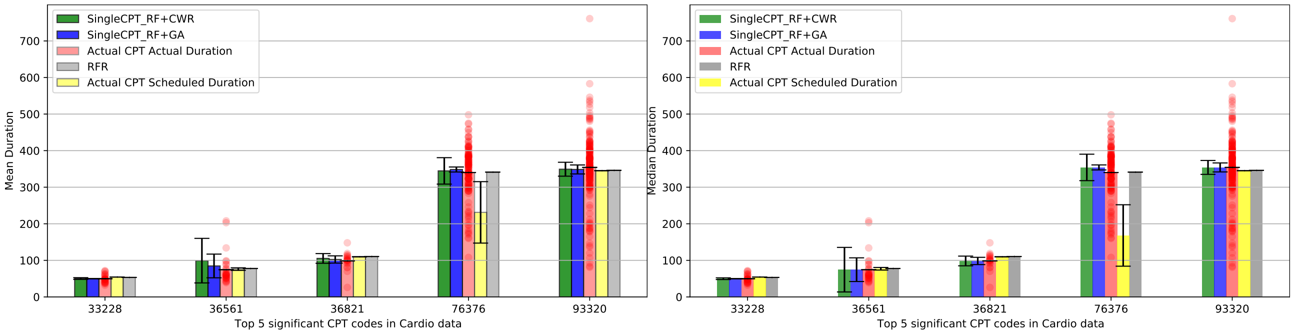
$$e_{d_i,d_i} = |Avg_{ActualCPT(ScheduledDuration)} - Avg_{ActualCPT(ActualDuration)}| \tag{17}$$

More precisely, under "CPT unknown" in Table 13, we reproduce the performance of state-of-the-art models and hospital surgical unit given the actual surgery duration of each surgery case. The hospital surgery unit performance is named as "Scha-Acta" and is based on the scheduled duration $(ScheduledDuration_i)$ and actual duration $(ActualDuration_i)$ of each surgery case (i). The error term for presented models under "CPT unknown" columns however, follow the equation (16) where the CPT is first predicted using two developed approaches and then surgery duration distributions are estimated. In equation (16) $Avg_{ActualCPT(ActualDuration)}$ represents the average duration of actual CPT's actual durations and likewise, $Avg_{PredCPT(ActualDuration)}$ represents the average duration of predicted CPT's actual durations in the training dataset. Therefore, if the predicted CPT matches the actual CPT the error will be zero. This implies that the accuracy of CPT prediction comes first when trying to predict the durations. We developed CWR model to improve the Random Forest classification in terms of CPT accuracy; however, in order to improve the duration estimation further we developed Random Forest classification with revised loss which contributes to produce even better durations. As duration results in Table 13 and CPT prediction accuracy in Table 10 suggest, the presented models are capable of not only producing reliable CPT but also a more credible and explainable duration estimation per case. This enables the hospital to calculate the duration distribution from actual durations in historical data by filtering the CPTs.

		CPT Known					CPT Unknown				
		Adj R-squared (%)	MSE (min)	MAE (min)	RMSE (min)	e_STD (min)	Adj R-squared (%)	MSE (min)	MAE (min)	RMSE (min)	e_STD (min)
Cardio	SVR	78	894	15	30	26	36	3627	38	60	47
	DTR	72	1074	23	33	24	40	3207	37	57	43
	RFR	88	541	13	23	19	49	2801	29	53	42
	MLP	88	505	12	22	19	53	2647	34	51	39
	Scha-Acta	88	528	13	23	20	51	2708	34	52	40
	RF						59	2473	21	50	44
	RF + CWR						61	2027	15	44	42
	RF + Revised Loss						**68**	**1551**	**12**	**39**	**37**
General	SVR	65	332	10	18	16	22	1481	25	38	29
	DTR	58	468	14	22	17	22	1484	25	39	30
	RFR	71	259	7	16	14	30	1323	18	36	27
	MLP	71	260	8	16	15	34	1270	25	36	26
	Scha-Acta	67	300	9	17	15	28	1398	25	37	28
	RF						51	664	10	26	22
	RF + CWR						56	576	8	24	22
	RF + Revised Loss						**57**	**518**	**8**	**22**	**20**
OBGYN	SVR	89	278	9	17	14	53	1906	28	44	34
	DTR	84	517	16	23	17	51	1940	28	44	33
	RFR	93	219	8	15	13	60	1602	21	40	32
	MLP	92	226	8	15	13	58	1717	27	41	31
	Scha-Acta	90	241	9	16	13	52	1927	28	44	34
	RF						66	1304	15	36	34
	RF + CWR						70	1197	13	35	30
	RF + Revised Loss						**70**	**1175**	**13**	**34**	**30**
Urology	SVR	84	166	7	13	11	62	762	18	28	22
	DTR	80	306	12	17	13	60	797	19	28	22
	RFR	85	109	6	10	9	68	663	13	26	20
	MLP	86	103	6	10	9	69	632	18	25	19
	Scha-Acta	84	136	8	12	9	67	699	18	26	19
	RF						79	478	9	22	19
	RF + CWR						80	303	8	17	15
	RF + Revised Loss						**81**	**286**	**7**	**17**	**14**
Other	SVR	50	650	15	25	21	29	1944	28	44	34
	DTR	48	794	19	28	22	18	2417	30	49	39
	RFR	53	597	14	24	20	23	2101	27	46	35
	MLP	59	495	14	22	18	35	1757	28	42	32
	Scha-Acta	53	595	15	24	19	31	1887	28	43	33
	RF						36	1711	23	41	33
	RF + CWR						47	935	15	30	26
	RF + Revised Loss						**49**	**801**	**14**	**28**	**24**

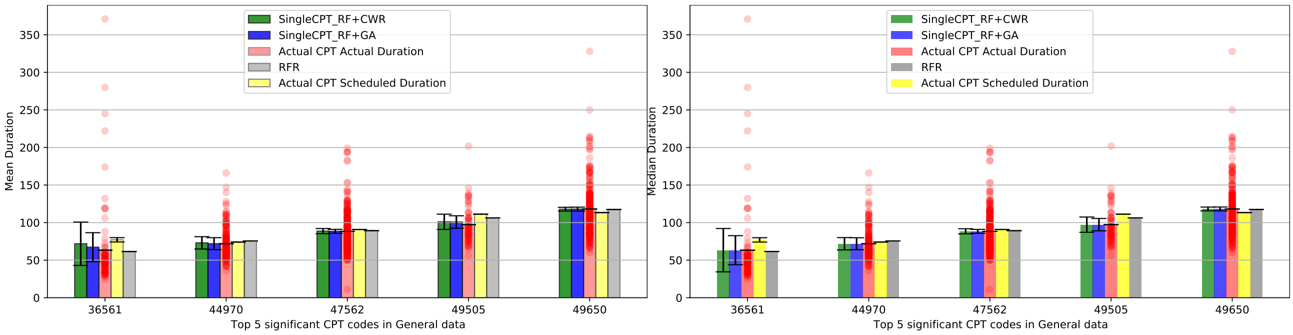
Table 13. Duration estimation performance in different specialty datasets (point estimates vs duration distributions) for CPT known assumption and when the CPT is unknown and predicted in our presented models.

Moreover, the duration distribution characteristics of top 5 CPT codes (with respect to frequency) are represented in terms of mean, median, and standard deviation for 5 specialties in Figures 26-30. We have compared the distribution statistics of presented models against the best performing traditional regression model; Random Forest Regression.



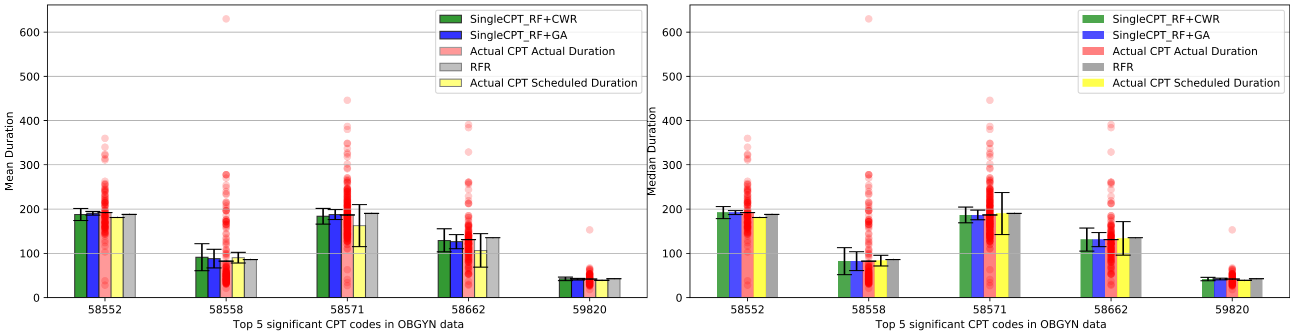
(a) Mean Duration (b) Median Duration

Figure 26. Model comparison: plot of mean, median and standard deviation of actual durations for top 5 CPTs in Cardio specialty.



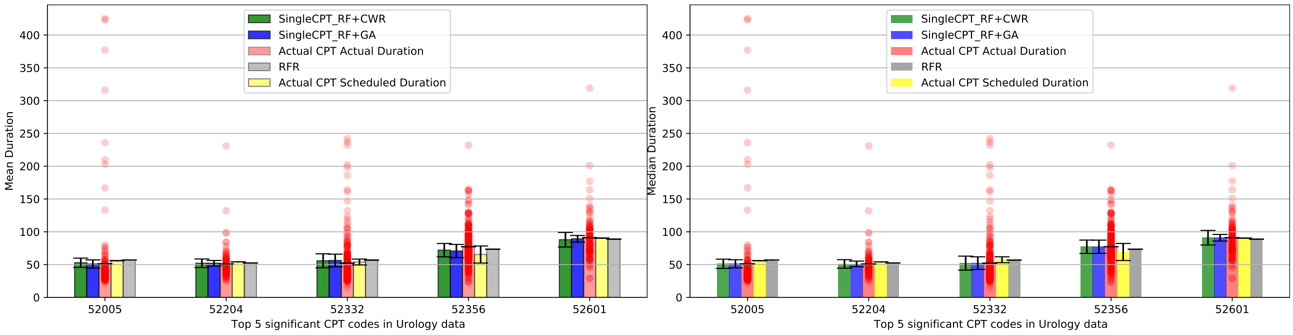
(a) Mean Duration (b) Median Duration

Figure 27. Model comparison: plot of mean, median and standard deviation of actual durations for top 5 CPTs in General specialty.



(a) Mean Duration (b) Median Duration

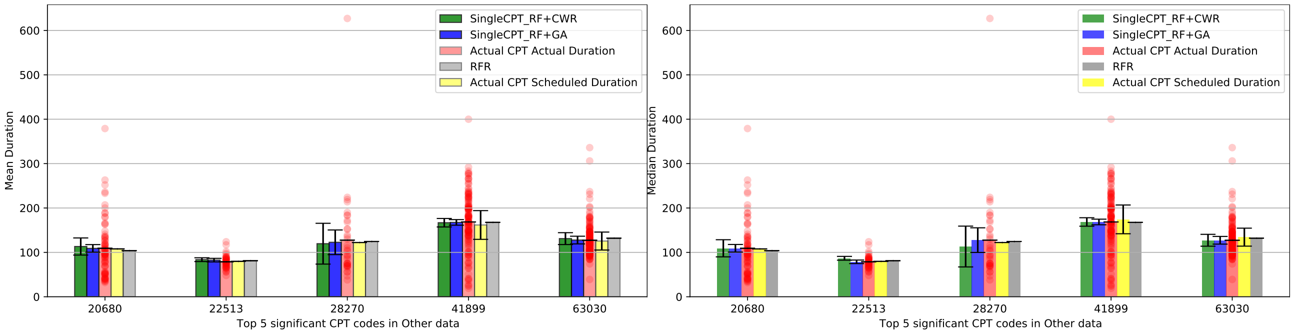
Figure 28. Model comparison: plot of mean, median and standard deviation of actual durations for top 5 CPTs in OBGYN specialty.



(a) Mean Duration

(b) Median Duration

Figure 29. Model comparison: plot of mean, median and standard deviation of actual durations for top 5 CPTs in Urology specialty.



(a) Mean Duration

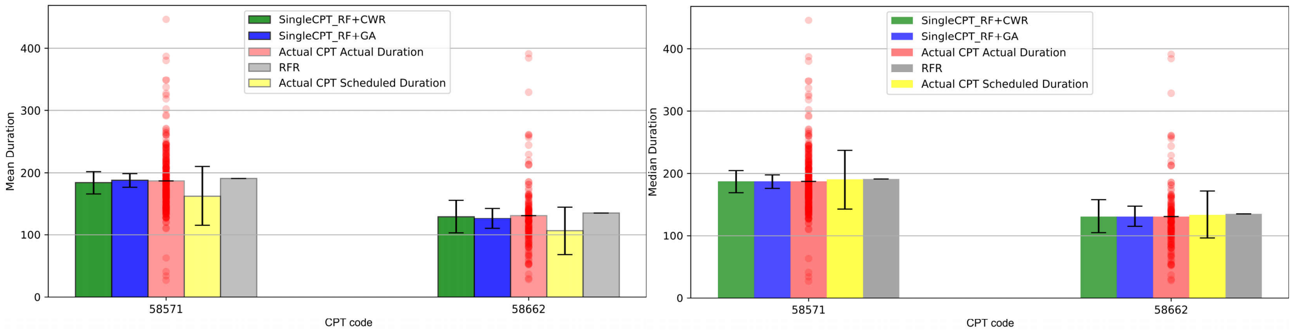
(b) Median Duration

Figure 30. Model comparison: plot of mean, median and standard deviation of actual durations for top 5 CPTs in Other specialty.

In Figures 26-30, we plot the duration distribution characteristics for CPT codes with highest number of label occurrences in each specialty dataset. These CPTs are known as the most significant CPTs in each specialty. While some of other CPT labels partially share same procedure information with significant CPTs, minor procedure variations appear in one or more additional tasks given specific circumstances such as patient age, urgent diagnostic matters at the time of surgery, etc. In sub-figures (a) (on the left hand side of Figures), the height of the bar reflect the average duration for our models and average scheduled and actual duration of actual CPT labels, while they show the median durations in sub-figures (b) for the same set of models.

By reviewing and comparing the sub-figures (median and mean durations) of CWR and GA models (green and blue bars), we observe that when the predicted mean and median durations deviates considerably, the median plot suggests more precise estimates with respect to the median actual duration bar (red bar) which indicates the model target. Then, the distribution recommendation can vary given how wide the standard deviation is in the results of our models.

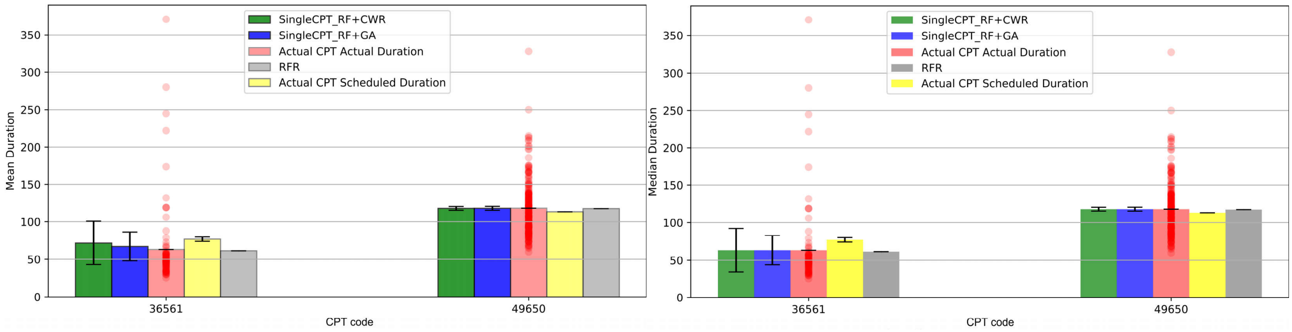
The wider the predicted duration standard deviation is the more likely the median would be a closer match to the actual duration median. This is due to the existence of outliers in predictions based on CPTs which may through the average off and skew it. In Figures 31-33, we extract some examples from Figures 26-30 to illustrate and analyze such behaviors more precisely.



(a) Mean Duration

(b) Median Duration

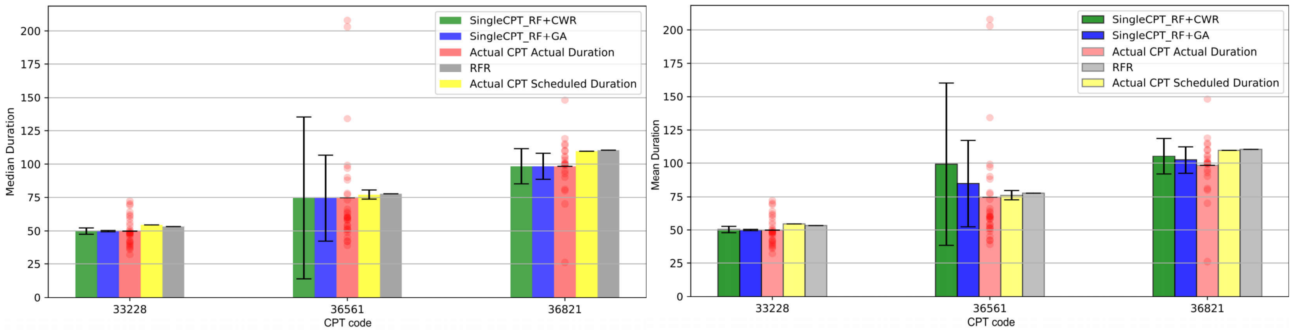
Figure 31. Model comparison: plot of mean, median and standard deviation of actual durations for the CPT durations with wide standard deviation in OBGYN specialty.



(a) Mean Duration

(b) Median Duration

Figure 32. Model comparison: plot of mean, median and standard deviation of actual durations for top 5 CPTs in Other specialty.



(a) Mean Duration

(b) Median Duration

Figure 33. Model comparison: plot of mean, median and standard deviation of actual durations for top 5 CPTs in Other specialty.

In Figure 31, CPTs "58571" and "58662" in OBGYN specialty have relatively wide duration prediction standard error. The average actual durations for CPTs predicted with "RF + CWR" and "RF + GA" models for "58571" and "58662" (as actual CPT labels in the models) are 183.9 and 185.2 minutes respectively, while the average actual duration for these CPTs are 186.6 and 131 minutes. Although, the average durations show that "RF + GA" approach outperform all other models including hospital's system performance, in median bar plot the median actual durations of predicted CPTs provide more promising duration estimates than average points. These phenomena are noticeable for other specialties such as General and Other; see Figures 32 and 33.

Lastly, K-S test (2 samples) is built upon two samples of durations: 1. The average actual duration of actual and predicted CPTs, and 2. The average scheduled duration of actual and predicted CPTs. The results show that, on average, 87% of the predicted and scheduled distributions of two samples are drawn from the same distribution.

To conclude, the presented two-step prediction methods (CWR or Revised Loss) are considered to produce more reliable results over the traditional (direct) methods. Providing a large feature set traditional machine learning models brings high complexity to the learning process. Therefore, the presented models reduce the duration prediction complexity by predicting the CPT labels first. Also, given that the surgeries are coded (CPT code) based on what subprocedures are performed in the surgery room, the results of CPT-based duration estimation models are more explainable and reliable. In other words, the distribution information through CPTs from indirect methods (over the point estimate based direct method) can be explained easily to the model users through the historical data and also introduce more reliable results for scheduling purposes.

6 Conclusion

6.1 Text Mining: Misspelling Correction and Abbreviation Detection in Healthcare

In the first step of this research, an unsupervised text mining algorithm, Hierarchical Agglomerative Clustering (HAC), combined with the most proper distance measure, Levenshtein distance (LD), and NLP methods have been proposed to correct the typos and detect abbreviations of the medical terms in the problem of free-text noisy surgery descriptions. The proposed approach permits finer data acquisition in an automated fashion. The primary application of this automated, robust, and yet highly reliable approach is improving unique word searching and highlighting in medical context to help the user ot quickly focus on retrieved important text information for further statistical analysis. In the process flow of this analysis, the Levenshtein distance matrix is furthered empowered with a capability to improve the distance of the pairs of surgical terms e.g. surgical terms and abbreviations, and surgical terms and typos.

The application of HAC method needs a rigid method of cutting at the right level of dendrogram to extract the most accurate clusters. Thus, the cluster weight metric is developed to heuristically find the best level of dendrogram given the related medical text and dendrogram characteristics. The cluster results produced by HAC, proposed LD, and cluster weight metric represent the sub-optimal cluster outcomes with respect to performance metrics. Hence, Heuristic clustering of HAC (HCHAC) approach is developed to find some clusters with false negatives members. The overall purpose of this phase is to provide an indicative representation of pruned and corrected text for next step research schema, CPT prediction and duration estimation.

In text mining area, the potential extension of the current method can be including coexistence features to improve both clustering results and CPT prediction. The combination of terms' occurrences in the surgery procedure descriptions can be a helpful factor in establishment of the CPT prediction model. It can also be experimented that the weight assigned to the distance between terms and their abbreviation forms changes at each iteration of the Hierarchical model in order to reduce the sparse clusters in the clustering results. Furthermore, the future upgrades of this research may offer a recommendation tool for text entries of the hospitals to reduce the noise in text entered by users in future surgical procedures.

6.2 Prediction: Surgery CPT Code Prediction, single and multiple CPT(s)

In this step of the research, the focus is utilizing and developing machine learning models / algorithms to classify surgery cases by primary or primary and auxiliary CPT codes. The dedication to improving medical text clusters in previous step help us apply what we've learned from text similarities to enhance the classification flow of CPT codes where we introduced the Class Weight Recalculation (CWR) algorithm. In our CPT classification analysis, while we have obtained encouraging primary CPT prediction accuracy results from Random Forest and CWR algorithm, the duration estimation performance with respect to mean duration prediction has not been as strong, especially when compared with those mean durations in direct method or OR scheduling system of the hospital.

Also, we observe high variations in CPT prediction and consequently mean durations in the distribution of many CPT cases. Although, the intermediate goal of CPT code(s) prediction is the classification accuracy, the ultimate goal is to utilize these code(s) predictions in characterizing duration distributions. Hence, in our proposed Random Forest model we incorporate duration mistmatch cost sensitivity to the CPT prediction (Objectives 2b and 3a). In other words, the Random Forest classification model utilized in this study is re-purposed to account for both duration loss (absolute error of duration prediction) and CPT classification loss (Gini index at each tree node) with coefficients that work as weight of each perspective. Whereas this reduced the CPT prediction accuracy, we observed that the cost sensitivity term with a proper choice of weight parameter can in fact improve the duration performance of the predicted CPT code(s) without much sacrifice from the classification accuracy.

In another stream of this research, the goal is predicting primary CPT and auxiliary CPT codes which represent the procedures in a surgery case collectively (rather than predicting only primary CPT). Predicting secondary CPTs in addition to primary CPTs provide more precise and helpful guidelines for operations planning. This also has the potential to improve the precision of surgery duration estimation where the second CPT contributes to durations considerably and is replicated through data points so the model can differentiate the code as primary and secondary more clearly. To achieve these goals, we proposed a deep neural multi channel model by employing the embedding layers for the text information and additional layers for bringing in the categorical features. We predict at 2 dominant CPT labels for each surgery using this approach; primary and secondary. The order of CPT codes are determined by the probabilites of the output matrix and RVU importance criteria.

6.3 Surgery Duration Distribution Estimation

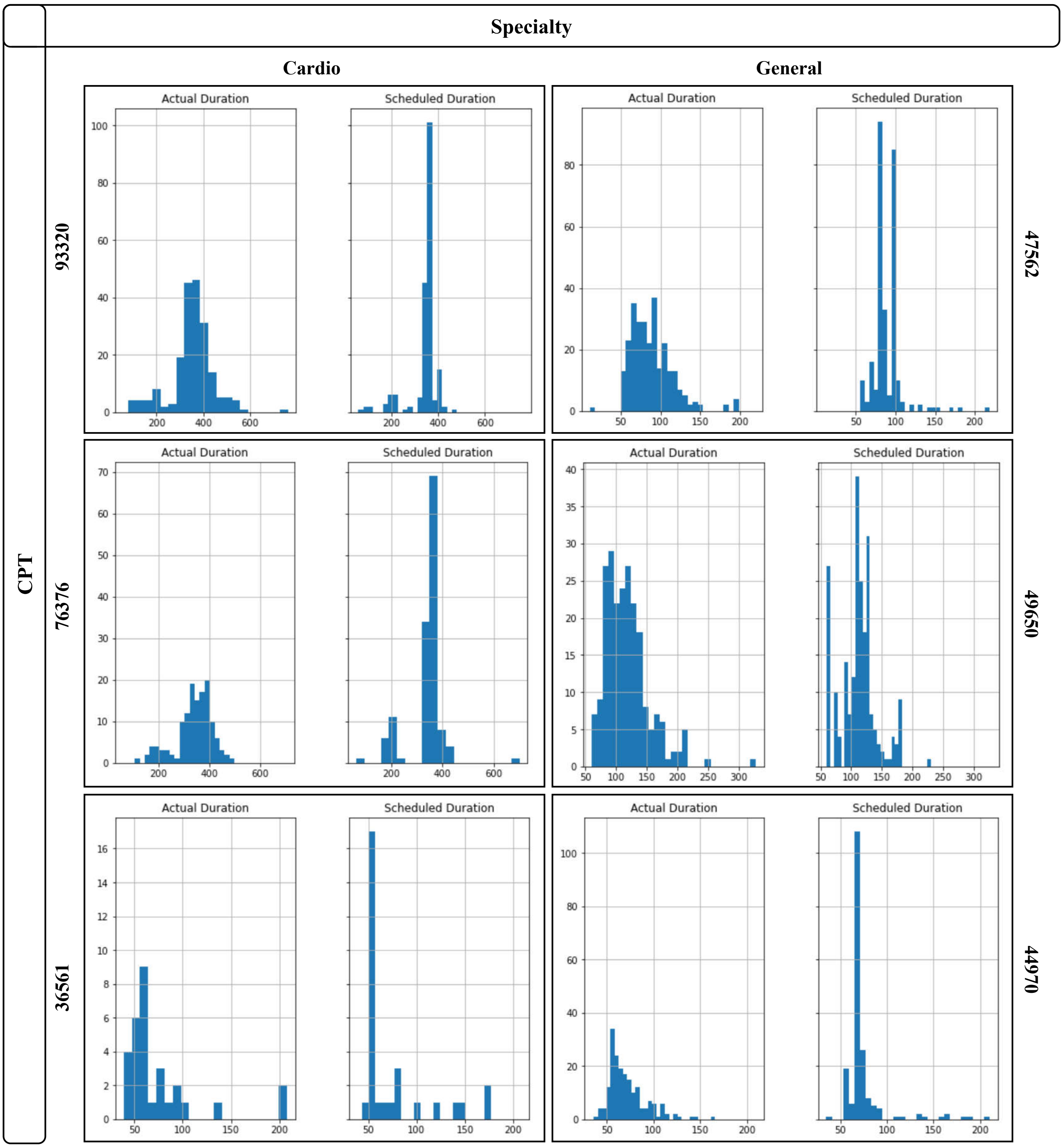
In this part of the remaining research, we evaluate the value of integrated CPT prediction and surgery duration prediction in improving the surgery duration objectives over the classical two-step approach. Towards this aim, we perform the following analysis:

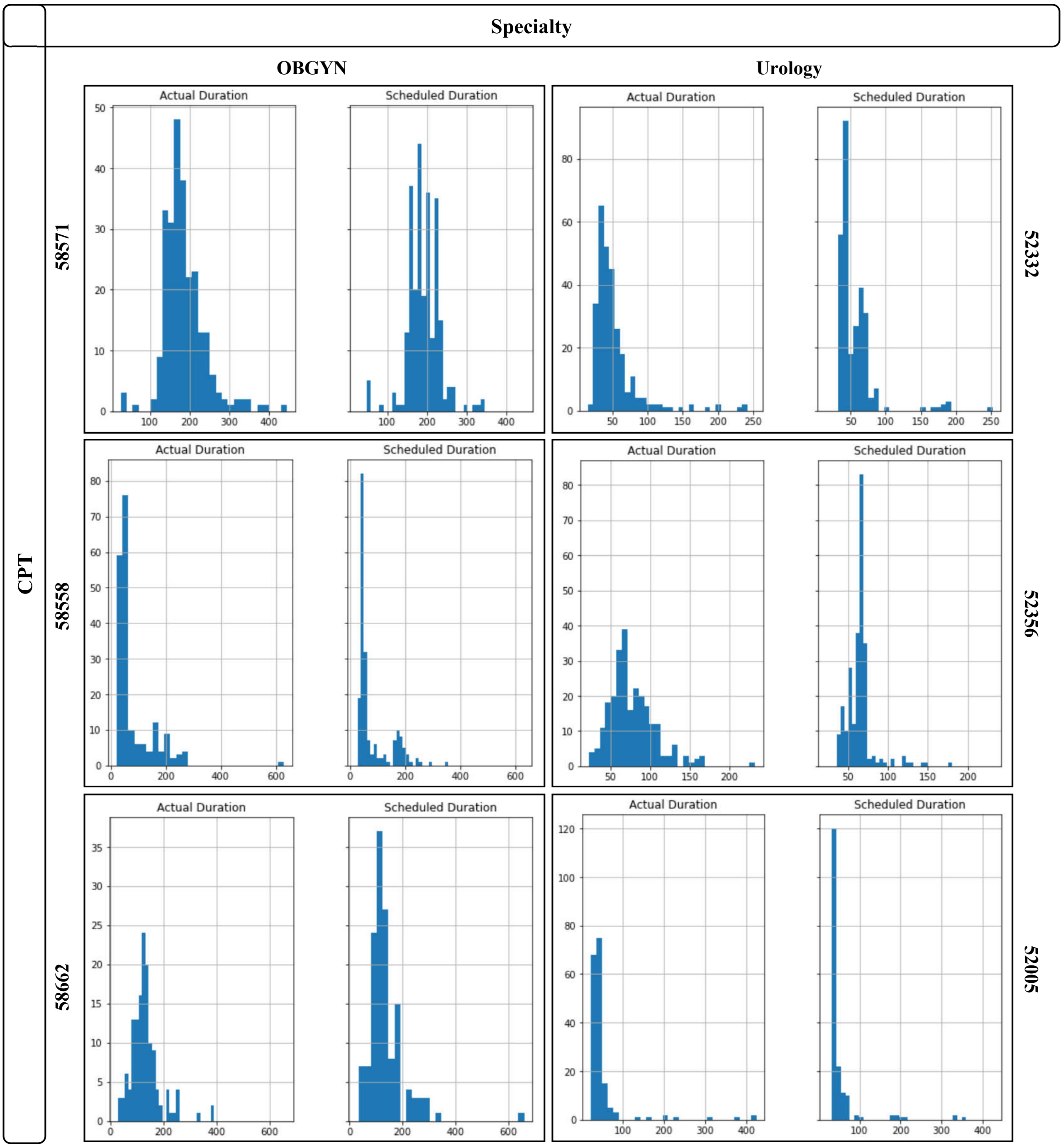
1. **RF + CWR:** Surgery duration distribution prediction using the primary CPT prediction where CWR method is used on top of RF predicted class probabilities to relearn the selective CPT code for each surgery case by focusing more on procedure text information.
2. **RF + Revised Loss using GA:** Surgery duration distribution prediction using the primary CPTs from RF model trees and optimize the selective subset of trees based on an objective function. The objective function takes into account both CPT prediction and duration estimation loss with respect to the optimized balancing coefficients.
3. **Multi Channel:** Predicting surgery durations using the primary and secondary CPTs predicted by multi-channel deep neural network model.

We then compare the performance of the proposed models to the preformance of the state-of-the-art regression models and the scheduling system of the hospital. The results show that the proposed models produce more reliable surgery duration distributions.

APPENDIX

Additional Surgery Duration Information




Specialty
OBGYN
Urology
CPT
58571
58558
58662
52332
52356
52005
Actual Duration
Scheduled Duration
0
10
20
30
40
50
60
70
80
100
120
5
15
25
35
150
200
250
300
400
600

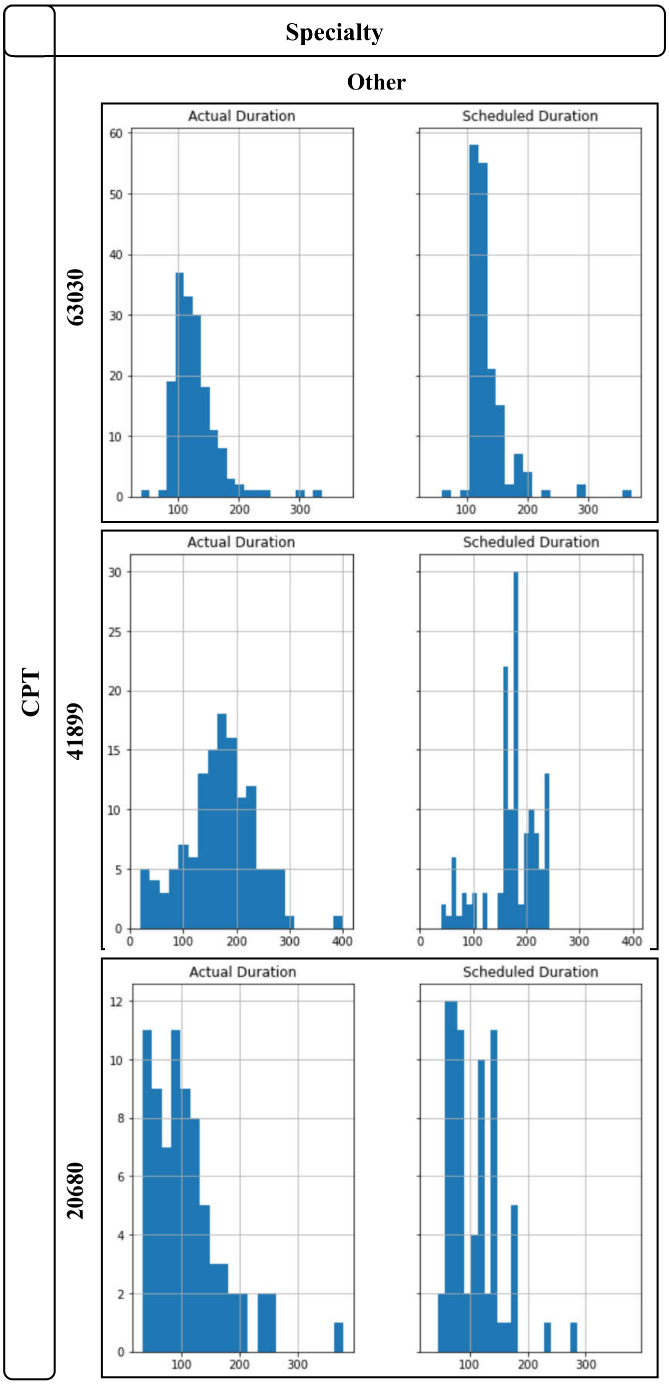


Figure 34. Actual and scheduled duration plots for significant CPTs in specialty datasets.

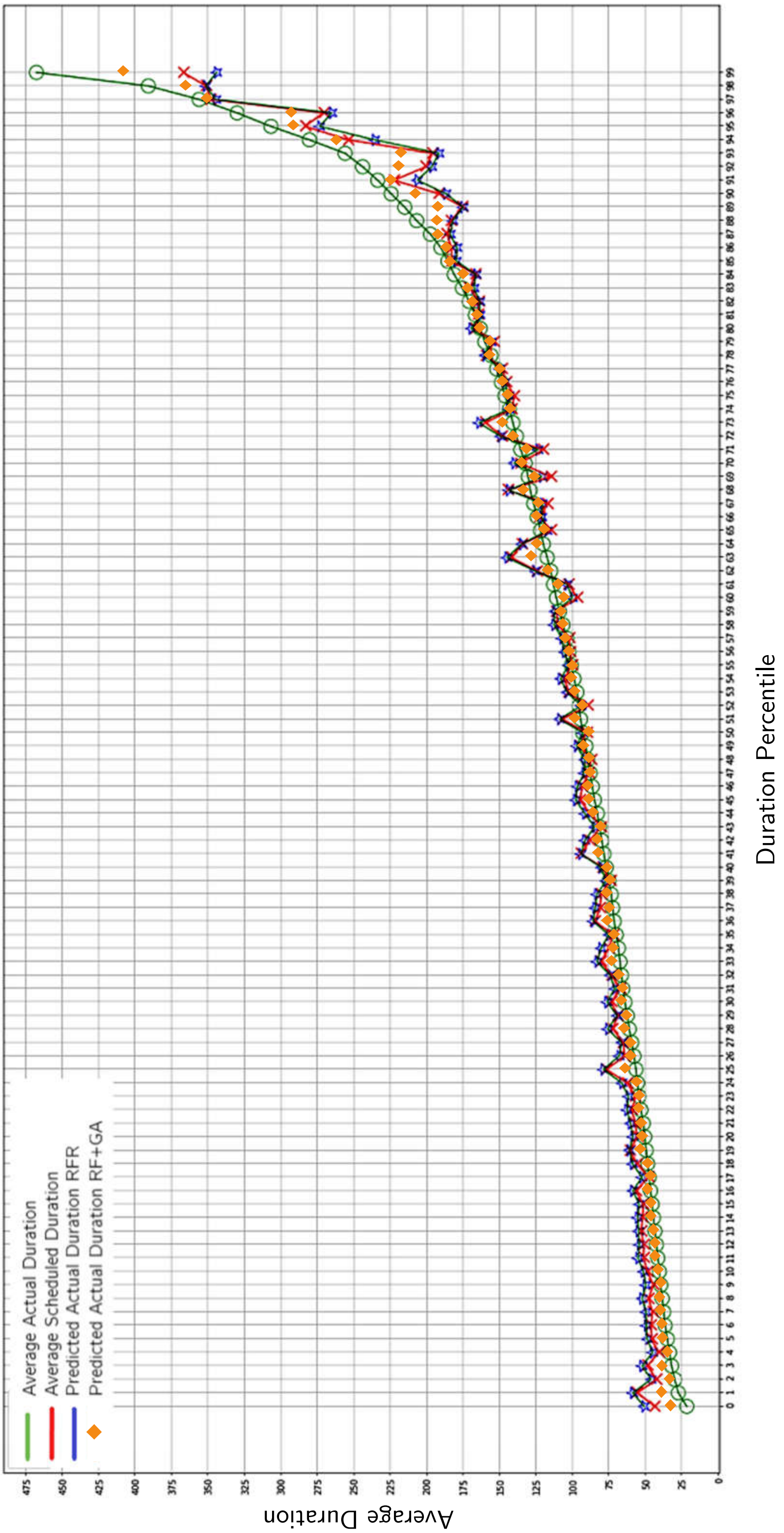
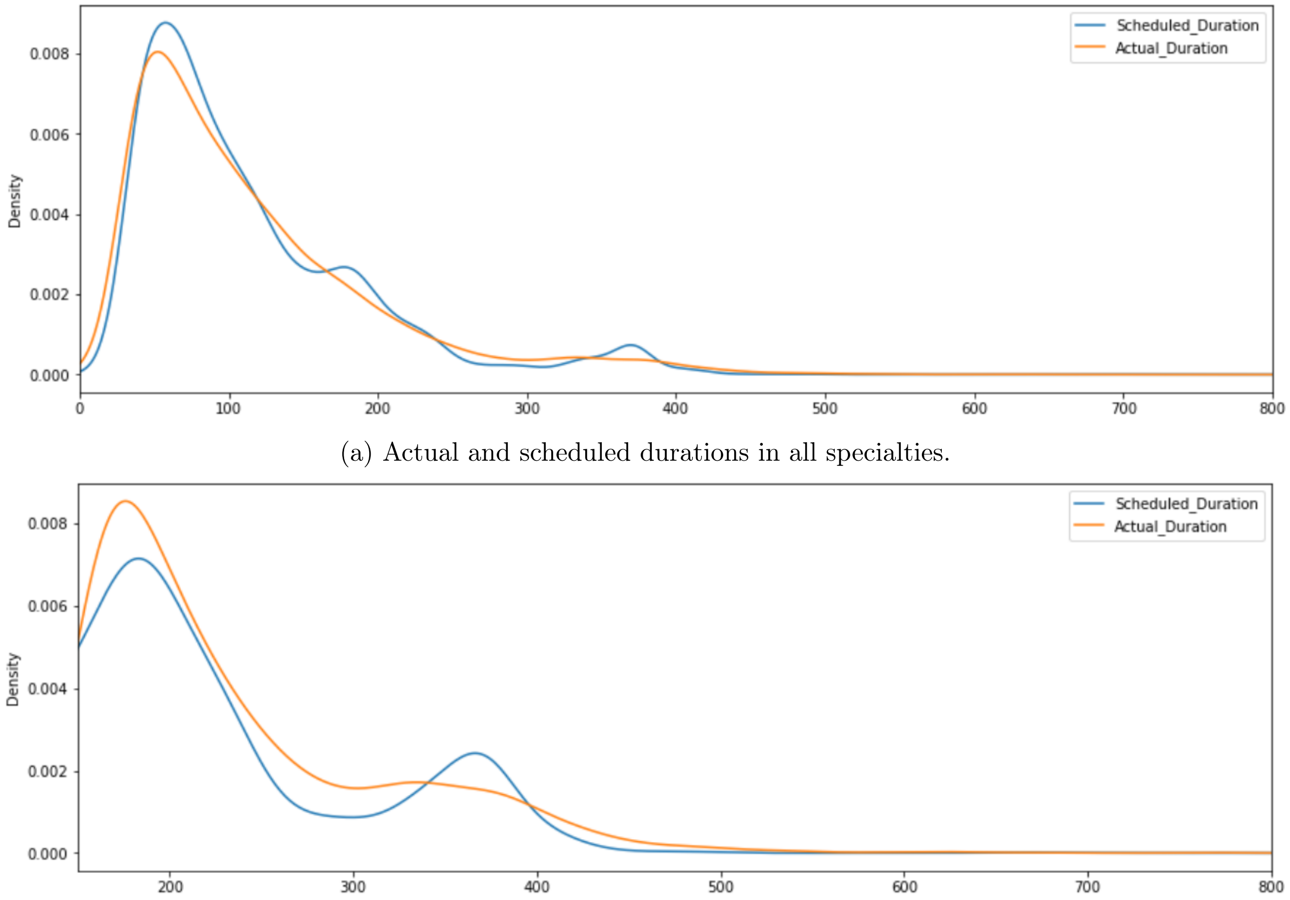


Figure 35. Duration percentile of average predicted and actual durations in all specialties.



(a) Actual and scheduled durations in all specialties.

(b) Actual and scheduled surgery durations greater than 150 minutes in all specialties.

Figure 36. Kernel Density Estimate plot using Gaussian kernels to estimate and show the probability density function (PDF) of both actual and scheduled durations.

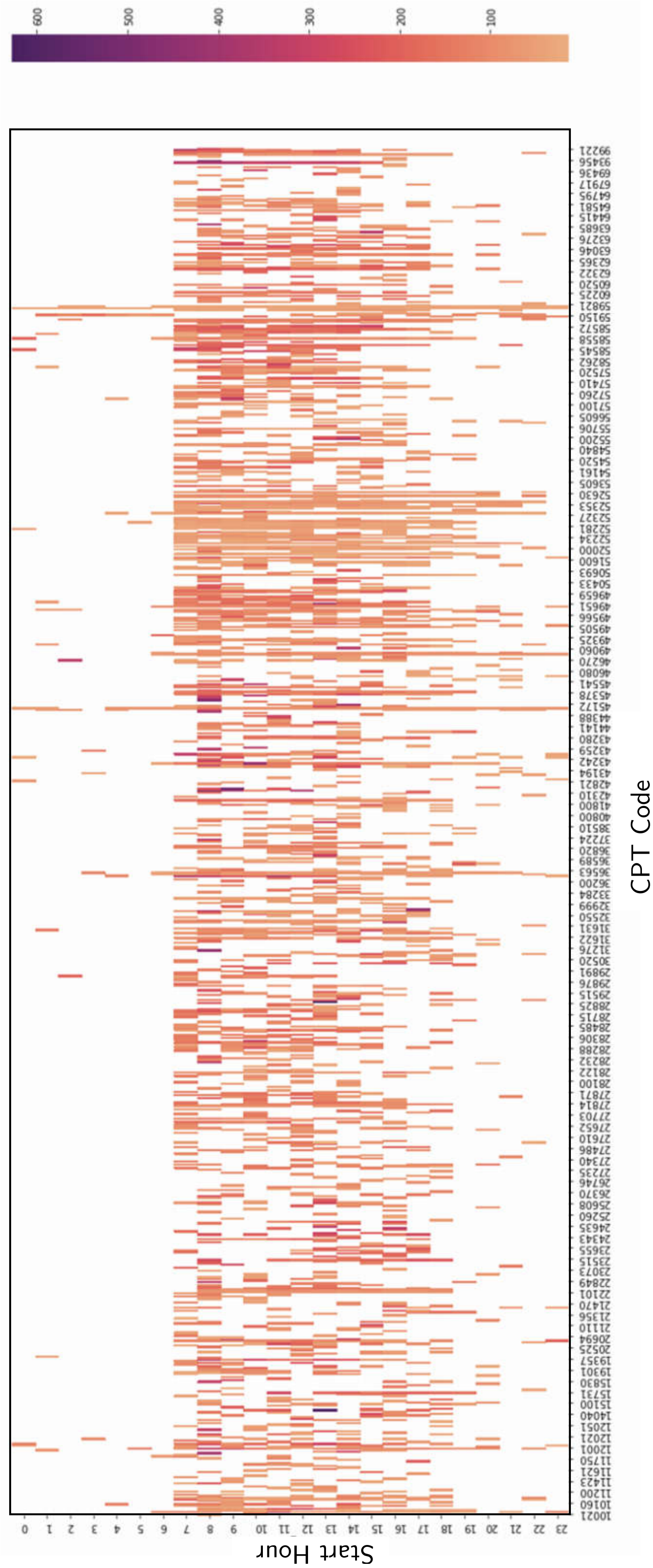


Figure 37. Heatmap of of average actual surgery duration per procedure start hour of surgeries (0-24) and CPT codes.

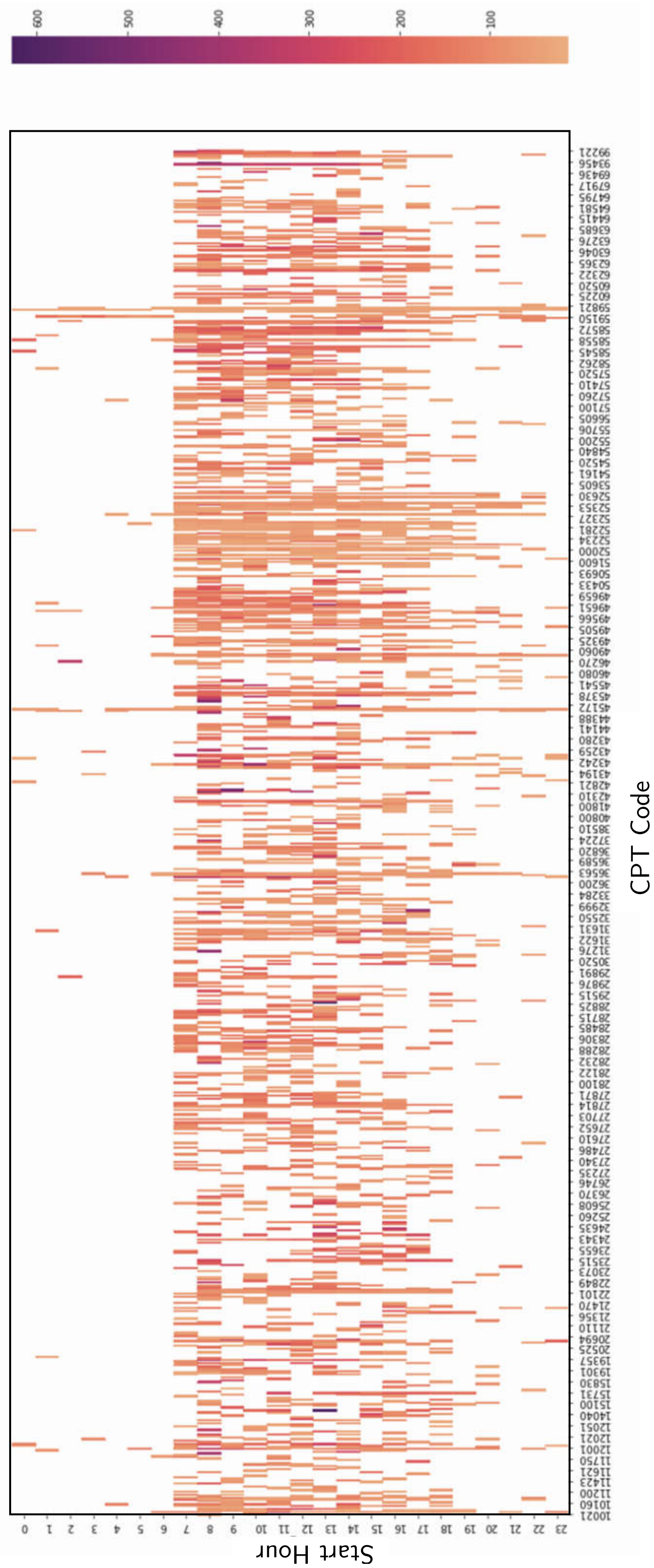


Figure 38. Heatmap of of average actual surgery duration specialty (pre-aggregation in terms of other) and CPT codes.

REFERENCES

[1] 8 medical coding mistakes that could cost you!

[2] Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Yangqing Jia, Rafal Jozefowicz, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dandelion Mané, Rajat Monga, Sherry Moore, Derek Murray, Chris Olah, Mike Schuster, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda Viégas, Oriol Vinyals, Pete Warden, Martin Wattenberg, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. TensorFlow: Large-scale machine learning on heterogeneous systems, 2015. URL https://www.tensorflow.org/. Software available from tensorflow.org.

[3] Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, et al. Tensorflow: Large-scale machine learning on heterogeneous distributed systems. *arXiv preprint arXiv:1603.04467*, 2016.

[4] Charu C Aggarwal and ChengXiang Zhai. *Mining text data*. Springer Science & Business Media, 2012.

[5] Mehdi Allahyari, Seyedamin Pouriyeh, Mehdi Assefi, Saied Safaei, Elizabeth D Trippe, Juan B Gutierrez, and Krys Kochut. A brief survey of text mining: Classification, clustering and extraction techniques. *arXiv preprint arXiv:1707.02919*, 2017.

[6] Kellie J Archer and Ryan V Kimes. Empirical characterization of random forest variable importance measures. *Computational Statistics & Data Analysis*, 52(4):2249–2260, 2008.

[7] Garth H Ballantyne, Douglas Ewing, Rafael F Capella, Joseph F Capella, Dan Davis, Hans J Schmidt, Annette Wasielewski, and Richard J Davies. The learning curve mea-

sured by operating times for laparoscopic and open gastric bypass: roles of surgeon's experience, institutional experience, body mass index and fellowship training. *Obesity surgery*, 15(2):172–182, 2005.

[8] Imon Banerjee, Yuan Ling, Matthew C Chen, Sadid A Hasan, Curtis P Langlotz, Nathaniel Moradzadeh, Brian Chapman, Timothy Amrhein, David Mong, Daniel L Rubin, et al. Comparative effectiveness of convolutional neural network (cnn) and recurrent neural network (rnn) architectures for radiology text report classification. *Artificial intelligence in medicine*, 97:79–88, 2019.

[9] Tal Baumel, Jumana Nassour-Kassis, Raphael Cohen, Michael Elhadad, and Noémie Elhadad. Multi-label classification of patient notes: case study on icd code assignment. In *Workshops at the thirty-second AAAI conference on artificial intelligence*, 2018.

[10] Federico Becattini, Tiberio Uricchio, Lorenzo Seidenari, Alberto Del Bimbo, and Lamberto Ballan. Am i done? predicting action progress in videos. *arXiv preprint arXiv:1705.01781*, 2017.

[11] Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5:135–146, 2017.

[12] Leo Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001.

[13] Adam Cannon, James Howse, Don Hush, and Clint Scovel. Learning with the neyman-pearson and min-max criteria. *Los Alamos National Laboratory, Tech. Rep. LA-UR*, pages 02–2951, 2002.

[14] Maria A Cassera, Bin Zheng, Danny V Martinec, Christy M Dunst, and Lee L Swanström. Surgical time independently affected by surgical team size. *The American journal of surgery*, 198(2):216–222, 2009.

[15] Tianqi Chen and Carlos Guestrin. Xgboost: A scalable tree boosting system. In *Proceed-*

ings of the 22nd acm sigkdd international conference on knowledge discovery and data mining, pages 785–794. ACM, 2016.

[16] Jung-Hsien Chiang, Jou-Wei Lin, and Chen-Wei Yang. Automated evaluation of electronic discharge notes to assess quality of care for cardiovascular diseases using medical language extraction and encoding system (medlee). *Journal of the American Medical Informatics Association*, 17(3):245–252, 2010.

[17] Billy Chiu, Gamal Crichton, Anna Korhonen, and Sampo Pyysalo. How to train good word embeddings for biomedical nlp. In *Proceedings of the 15th workshop on biomedical natural language processing*, pages 166–174, 2016.

[18] Catherine Combes, Nadine Meskens, Celine Rivat, and J-P Vandamme. Using a kdd process to forecast the duration of surgery. *International Journal of Production Economics*, 112(1):279–293, 2008.

[19] Glenn De'ath and Katharina E Fabricius. Classification and regression trees: a powerful yet simple technique for ecological data analysis. *Ecology*, 81(11):3178–3192, 2000.

[20] Franklin Dexter, Richard H Epstein, John D Lee, and Johannes Ledolter. Automatic updating of times remaining in surgical cases using bayesian analysis of historical case duration data and "instant messaging" updates from anesthesia providers. *Anesthesia & Analgesia*, 108(3):929–940, 2009.

[21] Franklin Dexter, Johannes Ledolter, Vikram Tiwari, and Richard H Epstein. Value of a scheduled duration quantified in terms of equivalent numbers of historical cases. *Anesthesia & Analgesia*, 117(1):205–210, 2013.

[22] Marinus JC Eijkemans, Mark Van Houdenhoven, Tien Nguyen, Eric Boersma, Ewout W Steyerberg, and Geert Kazemier. Predicting the unpredictablea new prediction model for operating room times using individual characteristics and the surgeon's estimate. *The Journal of the American Society of Anesthesiologists*, 112(1):41–49, 2010.

[23] Patrick Emami and Sanjay Ranka. Learning permutations with sinkhorn policy gradient. *arXiv preprint arXiv:1805.07010*, 2018.

[24] Martin Ester, Hans-Peter Kriegel, Jörg Sander, Xiaowei Xu, et al. A density-based algorithm for discovering clusters in large spatial databases with noise. In *Kdd*, volume 96, pages 226–231, 1996.

[25] Guozhong Feng, Jianhua Guo, Bing-Yi Jing, and Lizhu Hao. A bayesian feature selection paradigm for text classification. *Information Processing & Management*, 48(2):283–302, 2012.

[26] Stefan Franke, Jürgen Meixensberger, and Thomas Neumuth. Intervention time prediction from surgical low-level tasks. *Journal of biomedical informatics*, 46(1):152–159, 2013.

[27] Brendan J Frey and Delbert Dueck. Clustering by passing messages between data points. *science*, 315(5814):972–976, 2007.

[28] Carol Friedman, Lyudmila Shagina, Yves Lussier, and George Hripcsak. Automated encoding of clinical documents based on natural language processing. *Journal of the American Medical Informatics Association*, 11(5):392–402, 2004.

[29] Jerome H Friedman. Greedy function approximation: a gradient boosting machine. *Annals of statistics*, pages 1189–1232, 2001.

[30] Tushaar Gangavarapu, Aditya Jayasimha, Gokul S Krishnan, and Sowmya Kamath. Predicting icd-9 code groups with fuzzy similarity based supervised multi-label classification of unstructured clinical nursing notes. *Knowledge-Based Systems*, 190:105321, 2020.

[31] Brigid M Gillespie, Wendy Chaboyer, and Nicole Fairweather. Factors that influence the expected length of operation: results of a prospective study. *BMJ Qual Saf*, pages qhc–2011, 2011.

[32] Inmar E. Givoni, Clement Chung, and Brendan J. Frey. Hierarchical affinity propagation.

CoRR, abs/1202.3722, 2012. URL http://arxiv.org/abs/1202.3722.

[33] David E Goldberg and John Henry Holland. Genetic algorithms and machine learning. 1988.

[34] Wael H Gomaa and Aly A Fahmy. A survey of text similarity approaches. *International Journal of Computer Applications*, 68(13), 2013.

[35] Serkan Günal, Semih Ergin, M Bilginer Gülmezoğlu, and Ö Nezih Gerek. On feature extraction for spam e-mail detection. In *International Workshop on Multimedia Content Representation, Classification and Security*, pages 635–642. Springer, 2006.

[36] Fenfei Guo, Deqiang Han, and Chongzhao Han. k-intervals: a new extension of the k-means algorithm. In *Tools with Artificial Intelligence (ICTAI), 2014 IEEE 26th International Conference on*, pages 251–258. IEEE, 2014.

[37] Deepak Gupta, Barenya Bikash Hazarika, and Mohanadhas Berlin. Robust regularized extreme learning machine with asymmetric huber loss function. *Neural Computing and Applications*, 32(16):12971–12998, 2020.

[38] Zhe He, Zhiwei Chen, Sanghee Oh, Jinghui Hou, and Jiang Bian. Enriching consumer health vocabulary through mining a social q&a site: A similarity-based approach. *Journal of Biomedical Informatics*, 69:75–85, 2017.

[39] Wilbert Jan Heeringa. *Measuring dialect pronunciation differences using Levenshtein distance*. PhD thesis, University Library Groningen][Host], 2004.

[40] Anna Huang. Similarity measures for text document clustering. In *Proceedings of the sixth new zealand computer science research student conference (NZCSRSC2008), Christchurch, New Zealand*, pages 49–56, 2008.

[41] Zhexue Huang. Extensions to the k-means algorithm for clustering large data sets with categorical values. *Data mining and knowledge discovery*, 2(3):283–304, 1998.

[42] Zhenchao Jiang, Lishuang Li, Degen Huang, and Liuke Jin. Training word embeddings

for deep learning in biomedical text mining tasks. In *2015 IEEE international conference on bioinformatics and biomedicine (BIBM)*, pages 625–628. IEEE, 2015.

[43] Li-Ping Jing, Hou-Kuan Huang, and Hong-Bo Shi. Improved feature selection approach tfidf in text mining. In *Machine Learning and Cybernetics, 2002. Proceedings. 2002 International Conference on*, volume 2, pages 944–946. IEEE, 2002.

[44] Enis Kayis, Haiyan Wang, Meghna Patel, Tere Gonzalez, Shelen Jain, RJ Ramamurthi, Cipriano Santos, Sharad Singhal, Jaap Suermondt, and Karl Sylvester. Improving prediction of surgery duration using operational and temporal factors. In *AMIA Annual Symposium Proceedings*, volume 2012, page 456. American Medical Informatics Association, 2012.

[45] Tannaz Khaleghi, Alper Murat, and Hakimuddin Neemuchwala. Use of simulation in managing reusable medical equipment inventory in surgical services. In *Proceedings of the Summer Computer Simulation Conference*, page 39. Society for Computer Simulation International, 2016.

[46] Tannaz Khaleghi, Mohammad Abdollahi, and Alper Murat. Machine learning and simulation/optimization approaches to improve surgical services in healthcare. In *Analytics, Operations, and Strategic Decision Making in the Public Sector*, pages 138–165. IGI Global, 2019.

[47] Tannaz Khaleghi, Alper Murat, Suzan Arslanturk, and Eric Davies. Automated surgical term clustering: A text mining approach for unstructured textual surgery descriptions. *IEEE Journal of Biomedical and Health Informatics*, 2019.

[48] Youness Khourdifi and Mohamed Bahaj. Heart disease prediction and classification using machine learning algorithms optimized by particle swarm optimization and ant colony optimization. *International Journal of Intelligent Engineering & Systems*, 12(1):242–252, 2019.

[49] Yoon Kim. Convolutionalneuralnetworksforsentence classification.

[50] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[51] Christian Kruse, P Eiken, and P Vestergaard. Clinical fracture risk evaluated by hierarchical agglomerative clustering. *Osteoporosis International*, 28(3):819–832, 2017.

[52] Kenneth H Lai, Maxim Topaz, Foster R Goss, and Li Zhou. Automated misspelling detection and correction in clinical free-text records. *Journal of biomedical informatics*, 55:188–195, 2015.

[53] Eric Larsen, Sébastien Lachapelle, Yoshua Bengio, Emma Frejinger, Simon Lacoste-Julien, and Andrea Lodi. Predicting tactical solutions to operational planning problems under imperfect information. *arXiv preprint arXiv:1901.07935*, 2019.

[54] Miriam J Laugesen, Roy Wada, and Eric M Chen. In setting doctors' medicare fees, cms almost always accepts the relative value update panel's advice on work values. *Health affairs*, 31(5):965–972, 2012.

[55] Changhwan Lee, Yeesuk Kim, Young Soo Kim, and Jongseong Jang. Automatic disease annotation from radiology reports using artificial intelligence implemented by a recurrent neural network. *American Journal of Roentgenology*, 212(4):734–740, 2019.

[56] Fei Li and Hong Yu. Icd coding from clinical text using multi-filter residual convolutional neural network. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 8180–8187, 2020.

[57] Xinyu Li, Yanyi Zhang, Jianyu Zhang, Moliang Zhou, Shuhong Chen, Yue Gu, Yueyang Chen, Ivan Marsic, Richard A Farneth, and Randall S Burd. Progress estimation and phase detection for sequential processes. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 1(3):73, 2017.

[58] Ying Li, Saijuan Zhang, Reginald F Baugh, and Jianhua Z Huang. Predicting surgical

case durations using ill-conditioned cpt code matrix. *IIE Transactions*, 42(2):121–135, 2009.

[59] CH Bryan Liu, Benjamin Paul Chamberlain, Duncan A Little, and Ângelo Cardoso. Generalising random forest parameter optimisation to include stability and cost. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 102–113. Springer, 2017.

[60] Dong-sheng Liu and Shu-jiang Fan. A modified decision tree algorithm based on genetic algorithm for mobile user classification problem. *The Scientific World Journal*, 2014, 2014.

[61] Suliang Ma, Mingxuan Chen, Jianwen Wu, Yuhao Wang, Bowen Jia, and Yuan Jiang. Intelligent fault diagnosis of hvcb with feature space optimization-based random forest. *Sensors*, 18(4):1221, 2018.

[62] Marianne Maktabi and Thomas Neumuth. Online time and resource management based on surgical workflow time series analysis. *International journal of computer assisted radiology and surgery*, 12(2):325–338, 2017.

[63] Frank J Massey Jr. The kolmogorov-smirnov test for goodness of fit. *Journal of the American statistical Association*, 46(253):68–78, 1951.

[64] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, 2013.

[65] Agnieszka Mykowiecka and Małgorzata Marciniak. Domain-driven automatic spelling correction for mammography reports. In *Intelligent Information Processing and Web Mining*, pages 521–530. Springer, 2006.

[66] Seyed Amir Naghibi, Kourosh Ahmadi, and Alireza Daneshi. Application of support vector machine, random forest, and genetic algorithm optimized random forest models in groundwater potential mapping. *Water Resources Management*, 31(9):2761–2775, 2017.

[67] Feiping Nie, Zhanxuan Hu, and Xuelong Li. An investigation for loss functions widely used in machine learning. *Communications in Information and Systems*, 18(1):37–52, 2018.

[68] Alex Nowak, Soledad Villar, Afonso S Bandeira, and Joan Bruna. Revised note on learning quadratic assignment with graph neural networks. In *2018 IEEE Data Science Workshop (DSW)*, pages 1–5. IEEE, 2018.

[69] Randal S Olson and Jason H Moore. Tpot: A tree-based pipeline optimization tool for automating machine learning. In *Workshop on automatic machine learning*, pages 66–74. PMLR, 2016.

[70] M Paalvast, FC Meeuwsen, DMJ Tax, AP van Dijke, LSGL Wauben, M van der Elst, J Dankelman, JJ van den Dobbelsteen, et al. Real-time estimation of surgical procedure duration. In *E-health Networking, Application & Services (HealthCom), 2015 17th International Conference on*, pages 6–10. IEEE, 2015.

[71] Nicolas Padoy, Tobias Blum, Hubertus Feussner, Marie-Odile Berger, and Nassir Navab. On-line recognition of surgical activity for monitoring in the operating room. In *AAAI*, pages 1718–1724, 2008.

[72] Jon Patrick and Min Li. High accuracy information extraction of medication information from clinical notes: 2009 i2b2 medication extraction challenge. *Journal of the American Medical Informatics Association*, 17(5):524–527, 2010.

[73] Jon Patrick, Mojtaba Sabbagh, Suvir Jain, and Haifeng Zheng. Spelling correction in clinical notes with emphasis on first suggestion accuracy. In *Proceedings of 2nd Workshop on Building and Evaluating Resources for Biomedical Text Mining (BioTxtM2010)*, pages 1–8, 2010.

[74] Gary P Pisano, Richard MJ Bohmer, and Amy C Edmondson. Organizational differences in rates of learning: Evidence from the adoption of minimally invasive cardiac surgery.

Management Science, 47(6):752–768, 2001.

[75] Uzma Raja, Tara Mitchell, Timothy Day, and J Michael Hardin. Text mining in healthcare. applications and opportunities. *J Healthc Inf Manag*, 22(3):52–6, 2008.

[76] Karthik Ramasubramanian and Abhishek Singh. Deep learning using keras and tensorflow. In *Machine Learning Using R*, pages 667–688. Springer, 2019.

[77] Ellen Riloff, David Chiang, Julia Hockenmaier, and Jun'ichi Tsujii. Proceedings of the 2018 conference on empirical methods in natural language processing. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, 2018.

[78] Waheeda Saib, Tapiwa Chiwewe, and Elvira Singh. Hierarchical deep learning classification of unstructured pathology reports to automate icd-o morphology grading. *arXiv preprint arXiv:2009.00542*, 2020.

[79] Zahra ShahabiKargar, Sankalp Khanna, Abdul Sattar, and James Lind. Improved prediction of procedure duration for elective surgery. In *HIC*, pages 133–138, 2017.

[80] Patrice Y Simard, Yann A LeCun, John S Denker, and Bernard Victorri. Transformation invariance in pattern recognition—tangent distance and tangent propagation. In *Neural networks: tricks of the trade*, pages 239–274. Springer, 1998.

[81] William E Spangler, David P Strum, Luis G Vargas, and Jerrold H May. Estimating procedure times for surgeries by determining location parameters for the lognormal model. *Health care management science*, 7(2):97–104, 2004.

[82] Irena Spasić, Jacqueline Livsey, John A Keane, and Goran Nenadić. Text mining of cancer-related information: review of current status and future directions. *International journal of medical informatics*, 83(9):605–623, 2014.

[83] Pieter S Stepaniak, Christiaan Heij, Guido HH Mannaerts, Marcel de Quelerij, and Guus de Vries. Modeling procedure and surgical times for current procedural terminology-anesthesia-surgeon combinations and evaluation in terms of case-duration prediction and

operating room efficiency: a multicenter study. *Anesthesia & Analgesia*, 109(4):1232–1245, 2009.

[84] Pieter S Stepaniak, Christiaan Heij, and Guus De Vries. Modeling and prediction of surgical procedure times. *Statistica Neerlandica*, 64(1):1–18, 2010.

[85] David P Strum, Allan R Sampson, Jerrold H May, and Luis G Vargas. Surgeon and type of anesthesia predict variability in surgical procedure times. *Anesthesiology: The Journal of the American Society of Anesthesiologists*, 92(5):1454–1466, 2000.

[86] David P Strum, Jerrold H May, Allan R Sampson, Luis G Vargas, and William E Spangler. Estimating times of surgeries with two component procedurescomparison of the lognormal and normal models. *Anesthesiology: The Journal of the American Society of Anesthesiologists*, 98(1):232–240, 2003.

[87] Deliang Sun, Haijia Wen, Danzhou Wang, and Jiahui Xu. A random forest model of landslide susceptibility mapping based on hyperparameter optimization using bayes algorithm. *Geomorphology*, 362:107201, 2020.

[88] Hanna Suominen, Filip Ginter, Sampo Pyysalo, Antti Airola, Tapio Pahikkala, S Salanter, and Tapio Salakoski. Machine learning to automate the assignment of diagnosis codes to free-text radiology reports: a method description. In *Proceedings of the ICML/UAI/COLT workshop on machine learning for health-care applications*, 2008.

[89] Herman D Tolentino, Michael D Matters, Wikke Walop, Barbara Law, Wesley Tong, Fang Liu, Paul Fontelo, Katrin Kohl, and Daniel C Payne. A umls-based spell checker for natural language processing in vaccine safety. *BMC medical informatics and decision making*, 7(1):3, 2007.

[90] Elizabeth Travis, Sarah Woodhouse, Ruth Tan, Sandeep Patel, Jason Donovan, and Kit Brogan. Operating theatre time, where does it all go? a prospective observational study. *Bmj*, 349:g7182, 2014.

[91] Alper Kursat Uysal and Serkan Gunal. The impact of preprocessing on text classification. *Information Processing & Management*, 50(1):104–112, 2014.

[92] Gaetano Valenti, Maria Lelli, and Domenico Cucina. A comparative study of models for the incident duration prediction. *European Transport Research Review*, 2(2):103–111, 2010.

[93] Yefeng Wang. Annotating and recognising named entities in clinical notes. In *Proceedings of the ACL-IJCNLP 2009 Student Research Workshop*, pages 18–26. Association for Computational Linguistics, 2009.

[94] Richard A Wilson, Wendy W Chapman, Shawn J DeFries, Michael J Becich, and Brian E Chapman. Automated ancillary cancer history classification for mesothelioma patients from free-text clinical reports. *Journal of pathology informatics*, 1, 2010.

[95] Jingyi Xu, Zilu Zhang, Tal Friedman, Yitao Liang, and Guy Broeck. A semantic loss function for deep learning with symbolic knowledge. In *International Conference on Machine Learning*, pages 5502–5511. PMLR, 2018.

[96] Keyang Xu, Mike Lam, Jingzhi Pang, Xin Gao, Charlotte Band, Piyush Mathur, Frank Papay, Ashish K Khanna, Jacek B Cywinski, Kamal Maheshwari, et al. Multimodal machine learning for automated icd coding. In *Machine Learning for Healthcare Conference*, pages 197–215. PMLR, 2019.

[97] Xin Ye, Lu-an Dong, and Da Ma. Loan evaluation in p2p lending based on random forest optimized by genetic algorithm with profit score. *Electronic Commerce Research and Applications*, 32:23–36, 2018.

[98] Antoine E Zambelli. A data-driven approach to estimating the number of clusters in hierarchical clustering. *F1000Research*, 5, 2016.

[99] Qing T Zeng, Sergey Goryachev, Scott Weiss, Margarita Sordo, Shawn N Murphy, and Ross Lazarus. Extracting principal diagnosis, co-morbidity and smoking status for asthma

research: evaluation of a natural language processing system. *BMC medical informatics and decision making*, 6(1):30, 2006.

[100] Ning Zhong, Yuefeng Li, and Sheng-Tang Wu. Effective pattern discovery for text mining. *IEEE transactions on knowledge and data engineering*, 24(1):30–44, 2012.

ABSTRACT

IMPROVING OR OPERATIONS USING MACHINE LEARNING TECHNIQUES

by

TANNAZ KHALEGHI

December 2022

Advisor: Dr. Alper Murat

Major: Industrial & Systems Engineering

Degree: Doctor of Philosophy

Recently, health care related studies are being widely conducted by researchers using unique and efficient techniques to increase system profitability, quality of care, and patient satisfaction. Surgery department is considered as the hospital's engine, and cost of surgical services has a huge impact on the overall profitability of the hospital. This thesis proposes novel approaches to improve the efficiency of surgical services by using machine learning concepts.

In the first part, this research investigates the prediction of the surgery durations and Current Procedural Terminology (CPT) Codes. Accurate prediction of the surgery duration will improve the utilization of indispensable surgical resources such as surgeons, nurses, and operating rooms. Prediction of the correct CPT codes not only aids the preparation for the survery (i.e., case cart) but also enhances prediction of surgery duration distributions.

In predicting the CPT code(s) of each surgery, we use continuous, categorical and textual preoperative information as the independent features. Since information-rich textual information available perioperatively is mostly entered manually and thus is non-standardized (i.e. abbreviations) and prone to typos. Accordingly, direct usage of the raw text features leads to loss of text feature information. Thus, we first find the most informative text features from unstructured principal procedure and some physician notes through a novel text mining method for the detection and clustering of typos and abbreviations and efficiently reduces feature dimensionality. The output is a well-established in terms of typo correction and abbreviation detection and provides accuracy improvements in the prediction of CPTs as well as surgery durations. To predict CPTs, we first focus on the primary CPT prediction and evaluate the predictive performances of different filtering and set-based prediction strategies. While the primary CPT code is the most important determinant of surgery durations and periopera-

tive planning tasks, surgeries often entail multiple procedures (i.e., auxiliary CPTs) which can greatly influence the surgery durations. Hence, by using multi-task learning concepts, we develop models to predict multiple CPT codes, i.e. set containing the CPTs of all procedures being performed in the operation.

For the surgery duration prediction, we compare direct methods (i.e., regression based prediction using all feature information) with two-step approach where we first predict primary or set-CPTs of the surgery and then, given the predicted CPT codes, we estimate a duration distribution for each surgery case. By first predicting the CPT, the two-step approach provides valuable planning information to the preoperative services in addition to the improvements in surgery duration predictions. We evaluate the improvements in surgery duration estimation by comparing direct approach versus two-step approach and primary versus set-CPT predictions. Whereas direct approach primarily estimates the mean duration, the two-step approach naturally leads to a distribution information. We also evaluate the distributional information quality of the two-step approach with those that can be elicited from the direct approaches. Lastly, two-step approach also allows for more specific prediction and operational planning of surgical service operations such as case scheduling.

In order to account for the duration estimation loss in the single CPT prediction approach, we modified the CPT selection by applying the Genetic optimization algorithm. GA enables us to select the optimal trees with respect to the two goals, predicting correct CPT or estimating more accurate duration, in the model's boosting step. Lastly, we can compare the duration output of the revised single CPT approach with the aforesaid approaches. The hospital may choose to produce more accurate durations or weigh more on CPT prediction for the surgery cases given the package of CPT / duration prediction tool.

AUTOBIOGRAPHICAL STATEMENT

Tannaz Khaleghi is a Ph.D student of Industrial and Systems Engineering at Wayne State University, Detroit, Michigan. She received her Masters Degree in Industrial Engineering from the same university in 2015. While achieving her goals in academic world, Tannaz has worked on sophisticated data science projects for several years in leading companies in automotive industry. Her major research interests include Data Analytics, Healthcare Informatics, Machine Learning, Optimization and Supply Chain Management. Her research papers have been published in prestigious journals and conferences.