Assessing the effect of preprocessing of clinical notes on classification tasks and similarity measures

by

Md Moniruzzaman Moni

A Thesis Submitted to the Faculty of Graduate Studies of
The University of Manitoba
In partial fulfillment of the requirements of the degree of

MASTER OF SCIENCE

Department of Community Health Sciences
University of Manitoba
Winnipeg, Manitoba, Canada

Abstract

Background: Electronic Medical Records (EMRs), which contain both structured and unstructured information about a patient's medical history, are a rich source of data for population health and health services research. Unstructured text data (UTD) in EMRs may be challenging to use because of noise including spelling errors, abbreviations, and punctuation symbols. Preprocessing is potentially important to prepare UTD for research; it involves multiple steps. It is expected that preprocessing improves the performance of statistical or machine learning models.

Purpose and objectives: The research purpose was to examine the effect of preprocessing methods on analyses of UTD from EMRs. The objectives were to: (1) assess the effect of the number and order of preprocessing methods for UTD on the detection of health conditions, (2) assess the effect of the number and order of preprocessing methods on clinical and demographic cohort selection criteria in UTD, (3) assess the effect of the number and order of preprocessing methods on the similarity of information contained in pairs of EMR notes for the same patient, and (4) assess the effect of the number and order of preprocessing methods on accurate de-identification of UTD in EMR data.

Method: Study data were from the National Natural Language Processing Clinical Challenges, formerly known as Informatics for Integrating Biology and the Bedside (i2b2). Specifically, the 2008 i2b2 dataset was used for Objective 1, the 2018 i2b2 dataset was used for Objectives 2 and 3, and the 2014 i2b2 dataset was used for Objective 4. Preprocessing methods applied to the data included tokenization, removal of punctuation, correction of spelling errors, expansion of abbreviations, word stemming, and lemmatization. A nested experimental design was adopted, in which order was nested within number of methods. A balanced random forest model was used to detect 16 health conditions, and a support vector machine model was used to identify cohort

selection criteria for Objectives 1 and 2, respectively. A bidirectional long short-term memory-conditional random field model was used to de-identify UTD for Objective 4. Model performance was evaluated by measuring accuracy, sensitivity, specificity, F1 score, and precision for these objectives. For Objective 3, cosine similarity was used to measure the similarity of information between pairs of notes for the same patient. Analysis of Variance and descriptive statistics were used to test research hypotheses.

Results: Punctuation symbols comprised 18.5%, 19.0%, and 19.7% of the tokens in the 2008, 2018, and 2014 i2b2 datasets, while misspelled tokens comprised 5.3%, 6.3%, and 7.3% of the total tokens in these datasets. Tokens with abbreviations comprised 10.1%, 11.7%, and 12.4% of the total tokens in these three datasets, respectively. Mean model sensitivity ranged from 0.76 to 0.77, and mean specificity was 0.79 for detecting health conditions in Objective 1. The mean F1 score, accuracy, and precision increased by 1% using five preprocessing methods compared to using four preprocessing methods for Objective 1. Mean specificity, F1 score, accuracy, and precision ranged from 0.58 to 0.59, 0.48 to 0.49, 0.74 to 0.75, and 0.49 to 0.50, respectively for Objective 2. Cosine similarity scores were similar across preprocessing methods for Objective 3. Developed deep learning models for Objective 4 (de-identification) were not trainable with the preprocessed data. The results of ANOVA *F* tests showed no significant effect of the order of preprocessing for different numbers of methods. There was no difference in mean values of outcome variables among the number of methods.

Conclusion: Preprocessing reduced the size of each dataset. However, the order and number of preprocessing methods did not have any effect on model performance for a variety of tasks applied to text data. Future research could investigate the effect of the source of spelling correction libraries, medical dictionaries, and abbreviation lists on model performance results.

Acknowledgments

I would like to thank and acknowledge the support, guidance, and mentorship of my supervisor Dr. Lix during this thesis research. I would also like to thank my committee members for their key perspectives in this research, in particular Dr. Katz and Dr. Mohammed. This research was financially supported through the Canadian Institutes of Health Research (CIHR) funding to Dr. Lix. I am thankful to my colleagues at the Data Science Platform in the George and Fay Yee Centre for Healthcare Innovation and to my lab members for their support throughout my research project. Lastly and most importantly, I am truly grateful to my wife and baby, along with family and friends; thank you for your support.

Table of Contents

List of Tables

List of Figures

List of Abbreviations

Abbreviation	Definition
EMR	Electronic medical record
UTD	Unstructured text data
ICU	Intensive care unit
NLP	Natural language processing
NER	Named entity recognition
CRF	Conditional random field
WSD	word sense disambiguation
CPCSSN	Canadian primary care sentinel surveillance network
UCSF	University of California San Francisco
BIDMC	Beth Israel Deaconess Medical Center
TF	Term frequency
IDF	Inverse term frequency
TF-IDF	Term frequency-inverse document frequency
AUROC	Area under the receiver operating characteristic curve
POS	Part of speech
SS-Twitter	Sentiment Strength Twitter
SemEval	Semantic Evaluation
MIMIC	Medical information mart for intensive care
n2c2	National NLP Clinical Challenges
i2b2	Informatics for Integrating Biology and the Bedside
NIH	National institutes of health
PHI	Protected health information
CAD	Coronary artery disease, cardiovascular disease
HIPAA	Health Insurance Portability Accountability Act
Bi-LSTM	Bidirectional long short-term memory
CNN	Convolutional neural network
CHF	Congestive heart failure
DM	Diabetes mellitus

GERD	Gastroesophageal reflux disease
OSA	Obstructive sleep apnea
OA	Osteoarthritis
PVD	Peripheral vascular disease
ASP	Aspirin
MI	myocardial infarction
DIETSUPP	Dietary supplements,
HBA1C	Hemoglobin A1c
KETO	Ketoacidosis,
YR	Year
MOS	Months
NLTK	Natural language toolkit
TP	True positive
FP	False positive
FN	False negative
TN	True negative
RF	Random forest
BRF	Balanced random forest
SVM	Support vector machine
ANN	Artificial neural network
ANOVA	Analysis of variance
SD	Standard deviation
DF	Degrees of freedom

Chapter 1: Introduction

1.1 Background

Electronic medical records (EMRs) contain information about a patient's medical history, such as diagnoses, tests, medicines, allergies, immunizations, and treatments. EMRs allow healthcare providers to access patient information in a secure and timely manner and facilitate decisions about patient care, diagnoses, and treatment. EMRs are also a rich source of data for different types of studies such as longitudinal studies, cohort studies, case-control studies, and pragmatic clinical trials (1). However, the usefulness of EMR data for research depends on the structure, extractability, and quality of the data (1).

EMRs typically contain structured, semi-structured, and unstructured data (2). Structured data are coded, discrete, machine-readable fields. Many research studies use structured EMR data since it is often straightforward to process and analyze. Semi-structured data are a combination of structured and unstructured data and are more complex than structured data to process. These data are not in tabular format like structured data; they often have different formats. Unstructured EMR data do not have any specific formats; they include texts, images, audio recordings, and videos. Unstructured text data (UTD) are often in narrative form and can include healthcare provider notes, diagnostic reports, progress notes, surgical records, and discharge summaries. UTD from EMRs may contain information not captured in structured or semi-structured data (3). For example, patient characteristics extracted from UTD in EMRs to use as variables in risk prediction models have been shown to improve the performance of models to predict mortality in the intensive care unit (ICU) (4–6).

While UTD from EMRs has many uses for population health and health services research, these data can be challenging to use because of their high dimensionality, incompleteness, sparsity, heterogeneity, random errors, and presence of noise (7). The noise in

UTD includes spelling errors, symbols, and abbreviations (used instead of a word or a phrase) (8). These features of EMR data can lead to challenges when using natural language processing (NLP) and statistical text mining tools to extract features from UTD, such as diagnoses, risk factors, and treatments. Since UTD from EMR data may contain noise, they need to be effectively preprocessed and this can be a substantial task (9). The performance of statistical or machine learning models that are being used for the analysis of UTD may be affected by the quality of the data (10). Examples of UTD characteristics that have been used to measure data quality include the average length of a sentence, the percentage of abbreviations, and the percentage of spelling errors (11). High-quality data are essential to produce good analytic results.

Preprocessing of UTD is often the first stage in text analysis, and it often involves multiple steps to improve the overall quality of the data. Preprocessing involves preparing UTD for subsequent analysis, such as predictive model development (12). Different preprocessing methods may be adopted to transform UTD so that statistical or machine learning models can be applied. The preprocessing steps of a text analysis usually filter relevant texts, remove irrelevant words from the texts, and assign weights to relevant terms (13). Preprocessing methods, depending on the intended use of the data, may include tokenization, removal of stop words, removal of punctuation symbols, and word stemming. Preprocessing methods can be applied in different combinations and sequences.

Preprocessing is common in text classification tasks and may affect the performance of models that are used for classification (12). In previous studies in which different preprocessing methods were applied to UTD, preprocessing improved the performance of text classification

and sentiment analysis models (3,12–20). Preprocessing of short text also affects the measurement of similarity amongst text documents or strings (21).

However, there are no recommendations on the specific methods that should be adopted to preprocess UTD from EMRs (3), nor is there consensus on the order in which these methods should be applied to a dataset. There has been limited research on this issue. However, a study by Alnajran et al. (21) showed that the order of preprocessing methods affected the measurement of similarity amongst Twitter short text strings. Symeonidis et al. (18) showed that a pre-specified order of preprocessing methods improved the performance of sentiment analysis when they applied those methods to two datasets with three classes of positive, negative, and neutral sentiments.

1.2 Research objectives, questions, and hypotheses

The purpose of this research was to examine the effect of preprocessing methods on the analysis of UTD in EMRs. The objectives were to:

1) assess the effect of the order and number of preprocessing methods for UTD on detection of health conditions,
2) assess the effect of the order and number of preprocessing methods on clinical and demographic cohort selection criteria in UTD,
3) assess the effect of the order and number of preprocessing methods on the similarity of information contained in pairs of EMRs, and
4) assess the effect of the order and number of preprocessing methods on accurate de-identification of UTD in EMR data.

The following research questions were addressed:

1) Are the order and number of preprocessing methods associated with the correct detection of health conditions in UTD from EMRs, such as chronic diseases and their risk factors?
2) Are the order and number of preprocessing methods associated with the ability to identify clinical and demographic criteria for selecting cohorts using UTD in EMRs?
3) Are the order and number of preprocessing methods associated with the measurement of similarity in the content of EMRs for the same individual?
4) Are the order and number of preprocessing methods associated with the accurate removal of identifying information from UTD in EMRs?

The hypotheses were:

1) The order and number of preprocessing methods are associated with the correct detection of health conditions in EMRs.
2) The order and number of preprocessing methods are associated with the correct identification of cohort selection criteria in UTD from EMRs.
3) The order and number of preprocessing methods are associated with the estimated similarity of EMRs for the same individual.
4) The order and number of preprocessing methods are associated with the accurate de-identification of UTD from EMRs.

1.3 Thesis organization

The thesis is organized as follows: Chapter 2 provides a review of relevant literature about UTD data quality, preprocessing methods that may be applied to UTD, and the effect of preprocessing methods for UTD. This chapter concludes with a summary of research gaps. Chapter 3 provides information about the methods used in this research, including the study design, data sources, study variables, and descriptive and inferential analysis methods. Chapter 4

reports the effect of the order and number of preprocessing methods for classification tasks associated with Objectives 1 and 2. It contains estimates of model performance and the results of the inferential analyses. Chapter 5 reports the results for Objective 3, which focuses on similarity measures for EMRs with and without preprocessing the UTD. Chapter 6 discusses the challenges in implementing preprocessing methods when de-identification of UTD (i.e., Objective 4). Chapter 7 describes the key findings, discusses the research strengths and limitations, provides recommendations for future research, and provides conclusions drawn from this research.

Chapter 2: Literature Review

This chapter describes relevant literature on the assessment of data quality including definitions of data quality and the measure of UTD data quality. It provides an overview of the use of UTD in EMRs for research and describes research about preprocessing methods for UTD. A description of recent research with the i2b2 datasets, which were used to conduct this research and have been used in a variety of studies about the use of UTD from EMRs, follows. The chapter concludes with a summary of gaps in existing research about the effect of preprocessing methods applied to UTD in EMRs.

2.1 Data quality

Data quality assessment is a process to assess whether collected data are of the right type, quality, and quantity for the intended task (22). The ability of the collected data to meet the users' needs, which is known as fitness for use, is a common definition of data quality (23). The purpose of data quality assessment is to provide an evaluation and diagnosis of the data, including the data collection process, characteristics of the data, and the intended use of the data.

Data quality usually involves data preprocessing, profiling, and cleansing for subsequent tasks like data analyses (24). In the era of data-driven decisions, data quality assessment is an important part of the research process.

2.1.1 Definition of data quality

Data quality is a multi-dimensional construct about multiple features and characteristics of a dataset (25). Most data quality research has focused on the quality of structured data (26). Data quality encompasses correctness/accuracy, completeness, comparability, plausibility, currency, relevance, concordance, flexibility, usability/ease of use, security, and information loss and degradation (27). Data quality can be assessed in several ways, including by comparing to a

gold standard, data element agreement, element presence, distribution comparison, log review, conformance checks, qualitative assessment, and security analysis (27).

2.1.2 Measures of UTD data quality

Methods to assess the quality of UTD are often based on methods for structured data (28–30). UTD quality dimensions include interpretability, relevance, accuracy, readability, consistency, and accessibility (29,30). Quality indicators for UTD are similar to the indicators for structured data and are often used to assess the fitness of the data for statistical or machine learning models (10). The indicators of UTD quality include measures of noise, specificity, relevance, model fitting, and accuracy (28,30).

Noise in text data, including spelling errors, punctuation, symbols (other than alphanumeric characters), and abbreviations, makes UTD difficult for analyses (30). Preprocessing might increase the quality of UTD by reducing the amount of noise. There are rule-based and machine learning approaches in NLP that are used to assess the quality of text data. Rule-based methods are pattern-matching methods that involve setting hand-written rules for different purposes. For example, regular expressions, a rule-based approach, is used to find patterns in text data to capture punctuation symbols, special characters, URLs, email addresses, or abbreviations (31). The noise in the text data can be measured using the features that are captured by regular expressions, such as the percentage of punctuation symbols, abbreviations, special characters, and ungrammatical sentences (11). Other rule-based methods include sentence parsing, synonym extraction, Levenshtein distance metric, and Damerau-Leveshtein distance metric. Sentence parsing is a process of splitting out sentences into individual words assigned to different parts of speech using rules and patterns (32–34). Synonym extraction is used to extract similar meaningful words from the text documents (35). Rule-based methods are used to develop

a dictionary that contains synonyms of words. Levenshtein distance is a rule-based approach that measures the similarity between strings (36). Levenshtein distance is based on the number of edits needed to change one word to match a second word (e.g., Levenshtein distance 1 means the word requires one letter insertion, deletion or substitution from the original word). The Damerau-Leveshtein distance is similar to the Levenshtein distance, but it also considers the inversions between letters (e.g., words containing "ei" are mistakenly inversed as "ie") (36).

Machine learning approaches are probabilistic or classification models that describe the features of data and perform classifications or predictions based on the features of datasets (37). Machine learning models could be supervised or unsupervised. Some of the UTD quality indicators can be measured using machine learning models. For example, the Stanford named entity recognizer can be used to measure the percentage of abbreviations (11). This model is based on the conditional random field (CRF), which is a supervised machine learning model for sequential classifications. Homonym conflicts within text data can be detected and resolved by using word sense disambiguation (WSD) (28). This approach looks at the synonyms of words based on their context in the text. Latent semantic indexing, which is an unsupervised approach, is used as a computational framework for WSD.

2.2 Research using UTD in EMRs

The UTD from EMR is being used for epidemiological research of infectious diseases; for the surveillance of risk factors associated with acute and chronic diseases such as cancer, and diabetes; for estimation of incidence of different health conditions, or for clinical trial cohort selection (38). For example, features extracted from EMR text data are used in risk-prediction models (3–6). Researchers developed different classifiers to identify a range of procedures and diagnoses that predict ICU mortality using both UTD and structured data (4–6,38). Parreco et al.

(2018) used NLP techniques to process physician notes for surgical patients admitted to the ICU when developing a mortality risk prediction model (39). Sangaji et al. (2022) used rule-based methods to classify the most frequent diseases such as diabetes, hypertension, and renal failure using UTD from EMRs (40). They used an outpatient visit dataset that contained patient complaints, disease symptoms, and patient history. Kashina et al. (2020) used UTD from EMRs to check the effect of each stage of preprocessing on patients' allergic anamneses(12).

Canadian health researchers use UTD from EMRs for population health and health services research. The Canadian Primary Care Sentinel Surveillance Network (CPCSSN) maintains a database of information extracted from the EMRs of primary care physicians; it includes UTD that can be used for a variety of research projects. Garies et al. (2023) used pan-Canadian EMRs on antibiotics to validate a machine learning approach for cleaning and coding medication data (41). Meaney et al. (2022) used primary care clinical notes from EMR data to identify COVID-19 indicators using a rule-based NLP method (42). Another study used diagnostic text data from pan-Canadian medical records to identify posttraumatic stress disorder (43).

2.3 Research on preprocessing methods for UTD

Preprocessing methods are used to clean text and prepare it for analysis. There are various preprocessing methods available. Ha-Cohen-Kerner et al. (2020) divided preprocessing methods into two categories: (1) basic preprocessing methods, and (2) complex preprocessing methods (14). Basic preprocessing methods are primarily used to clean data and include conversion of uppercase font into the lowercase font, correction of spelling errors, removal of punctuation, removal of HTML tags, removal of stop words, and word stemming (i.e., reducing a word to its root form or stem). Complex preprocessing methods include word lemmatization

(reducing a word to its canonical or dictionary word), replacing slang with actual meaning, and expansion of abbreviations. The authors suggested using these methods to improve the quality of UTD for text classification tasks.

Kunilovskaya and Plum (2021) noted that data preprocessing refers to a wide range of methods, including cleaning (removing noise such as punctuation symbols), filtering, normalizing, and transforming raw data (44). They described a systematic approach to clean text data. This approach included an assessment of the structural completeness of the data by removing rows with missing information, cleaning the text by removing noise, and normalizing the textual data by expanding contractions for lemmatization. They focused on data for a Digital Humanities project from seven web aggregators of cultural information. The researchers found that preprocessing of text data reduced the size of the data by removing noise. The researchers showed that model performance for classification tasks improved when using the data after preprocessing compared to using raw UTD.

Ha-Cohen-Kerner et al. (2020) explored the impact of different combinations of preprocessing methods on text classification (14). The preprocessing methods included spelling correction, HTML tag removal, conversion of uppercase font to lowercase font, punctuation removal, reduction of repeated characters, and removal of stop words. The authors used three machine learning models to evaluate the influence of preprocessing methods on text classification. Overall, the authors found that the best results for text classification could be achieved with a combination of preprocessing methods, while removal of stop words could be very effective in improving model performance for text classification. However, no single combination of methods was found to be optimal, leading the authors to conclude that the choice of preprocessing methods depends on the characteristics of the dataset and the proposed analysis.

Dařena and Žižka (2015) assessed the impact of UTD preprocessing methods on the quality of the results and computational complexity of classification models (45). The data were collected from different hotels around the world and contained guest reviews written in English, German, Spanish, and French languages. They transformed the data into a sequence of words without ordering and a numerical vector to apply machine learning approaches. Three preprocessing methods, in different orders, were applied to the data: (i) correction of spelling errors using MS Word 2010 with misspelled words replaced by the alternative with the highest probability of being correct, (ii) word stemming based on Snowball (a programming language for stemming algorithms), and (iii) stop word removal using a general stop word list. The researchers found that the results of classification models were different for different languages using the same preprocessing methods. Model performance was not influenced significantly by the choice of preprocessing methods.

Mahendra et al. (2021) evaluated the impact of different preprocessing methods for clinical text data on the performance of machine learning models. The data were collected from ICUs at the University of California San Francisco (UCSF) and Beth Israel Deaconess Medical Center (BIDMC). The authors first cleaned the UTD by removing punctuation, numeric values, and stop words. Then they used the cleaned text for word stemming to shorten common words. The resulting stemmed text was transformed into term frequency-inverse document frequency (TF-IDF) vectors. TF-IDF vectors reflect important words in a collection of documents. TF-IDF vectors were created separately, using single stems only, and N-grams which were combinations of unigram (single word), bigram (two words), and trigram (three words) stems. UCSF data were used to train several models, including penalized logistic regression (i.e., Least Absolute Shrinkage and Selection Operator), random forest, and multilayer feed-forward neural network

models. Both UCSF and BIDMC data were used for model validation. The area under the receiver operating characteristic curve (AUROC) was used to assess model performance. The researchers found that the preprocessing pathway that included cleaning (i.e., removing punctuation symbols, numbers, and stop words), word stemming, and TF-IDF vectorization resulted in the greatest improvement in model performance compared to the preprocessing pathway that included cleaning, word stemming, and n-gram creation.

Kashina et al. (2020) checked the effect of preprocessing methods on the identification of patients' allergic anamneses (i.e., information about allergies and related information such as allergens) (12). The researchers used unstructured medical data provided by the Federal State Budgetary Institution in Russia. They sequentially used error correction (i.e. correcting typographical errors and spelling errors), deleting duplicate entries, removing characters and extra spaces, normalization (i.e., lemmatization), removal of stop words, and harmonization of classes (i.e., class balance) by creating synthetic data. The researchers evaluated the effectiveness of preprocessing methods in classifying medical text at every stage. The most effective preprocessing stage was performing normalization after error correction. Error correction and normalization resulted in improved model performance when compared to using raw data (i.e., data that were not preprocessed).

Few articles discuss the effect of the order of preprocessing methods, whereas several articles discuss the effect of the number of preprocessing methods and their combinations on text analyses. Alnajran et al. (2018) compared a heuristic-based preprocessing method to a conventional method to assess the impact of preprocessing methods on the performance of similarity measures for short texts from Twitter (21). The authors used different preprocessing methods in the following order to preprocess the short texts: decoding the text data (i.e.,

transforming the text into machine readable format), removing retweets and URLs, converting html characters to standard html tags (e.g., & is converted to and), tokenization, part of speech (POS) tagging, removing punctuation symbols except "?" and "!", lemmatization, replacing mentions of any user with only user, standardizing words and contractions, and splitting joint hashtags. Cosine similarity was used to assess the effect of preprocessing methods on similarity measures. They found that their proposed preprocessing methods with the specific order increased data quality in the context of similarity measures compared to the raw data and the conventional preprocessing methods.

Naseem et al. (2020) analyzed the effectiveness of improving text data quality by preprocessing for tweet classification. The study used three Twitter datasets on hate speech and both machine learning and deep learning methods for classification. The study used twelve preprocessing methods both separately and in combination: removal of Unicode, URLs, user-mentions, and hashtag symbols, replacing emoticons and emojis, replacing slang and abbreviations, correcting spelling errors, expanding contractions, replacing elongated words, removing punctuations, lowercasing of words, word segmentation, removing numbers, removing stop words, and lemmatization. The study found that the proposed order of twelve preprocessing methods improved model performance compared to other orders and numbers of methods.

Symeonidis et al. (2018) examined the significance of preprocessing methods in emotional text analyses (18). The researchers used 16 preprocessing methods and used existing Sentiment Strength Twitter (SS-Twitter) and Semantic Evaluation (SemEval) datasets and four supervised machine learning algorithms to measure accuracy. They found that replacing URLs and user mentions, replacing contractions, removing numbers, replacing repetitions of punctuations, and lemmatization resulted in the highest accuracy when the preprocessing

methods were applied independently. All 16 preprocessing methods and a combination of five methods were used for an ablation study (while using a machine learning system, a set of components is removed to measure the performance of the model to investigate the impact of the removed components). The results showed that the performance of a sentiment analysis was significantly improved when compared to an analysis based on raw data.

Table 2.1 summarizes the characteristics of studies that preprocessed UTD to improve data quality and model performance.

Table 2.1: Attributes of the studies addressing preprocessing methods of UTD.

Study	**Type of data**	**Preprocessing methods**	**Research findings**
Kunilovskaya and Plum (44)	Cultural news for digital humanities project	The following order of four preprocessing methods was applied to the data: removing rows with missing information, removing noise, expansion of contractions, and lemmatization.	Preprocessing reduced the size of the data and increased model performance for classification tasks.
Ha-Cohen-Kerner et al. (14)	Four datasets: documents of webpages, Reuters news articles, SMS spam messages, and reviews of products, movies, and restaurants	Sixty-three combinations of the following preprocessing methods were explored: spelling correction, HTML tag removal, conversion of uppercase font to lowercase font, punctuation removal, reduction of repeated characters, and removal of stop words.	The best results for text classification were achieved with a combination of preprocessing methods that might depend on the characteristics of the dataset and the proposed analysis.
Dařena and Žižka (45)	Hotel reviews	Different orders of three preprocessing methods were applied to the data, including correction of spelling errors, word stemming, and removal of stop words.	The model performance was not influenced significantly for the classification task.
Mahendra et al. (3)	ICU notes	Data were preprocessed using two preprocessing	The first preprocessing pathway improved the

		pathways: (1) removing punctuation symbols, numbers, and stop words, stemming, and TF-IDF vectorization, and (2) removing punctuation symbols, numbers, and stop words, stemming, and n-grams.	model performance compared to the second pathway.
Kashina et al. (12)	Medical records	The following methods were performed sequentially to preprocess the data: correcting typographical errors and spelling, deleting duplicate entries, removing characters and extra spaces, lemmatization, removal of stop words, and harmonization of classes.	The model performance improved compared to raw data when error correction and lemmatization were performed sequentially.
Alnajran et al. (21)	Short texts from Twitter	Ten preprocessing methods were applied in the following order: decoding the text data, removing retweets and URLs, converting HTML tags, tokenization, part of speech (POS) tagging, removing punctuation symbols except "?" and "!", lemmatization, replacing mentions, standardizing words, and contractions, and splitting joint hashtags.	The heuristic-based preprocessing methods improved the similarity of information compared to the raw data and the conventional preprocessing methods.
Naseem et al. (16)	Short texts from Twitter	Twelve preprocessing methods were applied separately and in combination: removal of Unicode, URLs, user-mentions, and hashtag	The proposed order of preprocessing methods improved the quality of short text by removing noise and the classification model performance

		symbols; replacing emoticons and emojis; replacing slang and abbreviations; correcting spelling errors; expanding contractions; replacing elongated words; removing punctuation; lowercasing of words; word segmentation; removing numbers; removing stop words; and lemmatization.	compared to other orders and numbers of methods.
Symeonidis et al. (18)	Short texts from Twitter	Sixteen preprocessing methods were applied to the data individually, and a combination of the best five methods. The preprocessing methods included removing Unicode strings and noise, replacing URLs and user mentions, replacing slang and abbreviations, replacing contractions, removing numbers, replacing repetitions of punctuation, replacing negations with acronyms, removing punctuation, handling capitalized words, lowercasing, removing stop words, replacing elongated words, spelling correction, POS tagging, lemmatization, and stemming.	Replacing URLs and user mentions, replacing contractions, removing numbers, replacing repetitions of punctuations, and lemmatization individually resulted in the highest accuracy for sentiment analysis and the combination of these methods also increased the performance of sentiment analysis.

2.4 Public datasets for research about UTD methods

There are limited resources of publicly available datasets of UTD from EMRs for research. The National NLP Clinical Challenges (n2c2) datasets and Medical Information Mart for Intensive Care (MIMIC) datasets of clinical notes are publicly available datasets. However, the n2c2, formerly known as i2b2 (Informatics for Integrating Biology and the Bedside), is the only data source that has annotated data. Researchers need to annotate the clinical notes collected from MIMIC datasets.

Many researchers use the i2b2 datasets to test methods for the analysis of UTD data. The i2b2 organizes national NLP clinical challenges on UTD collected from the partner's healthcare research patient data repository (46). The i2b2 arranged NLP shared task challenges beginning in 2006 on various health-related topics such as de-identification, smoking, obesity, medication, temporal relations, cohort selection, and adverse drug events. The i2b2 was funded by the US National Institutes of Health (NIH) and partnered with Boston's healthcare system. After each challenge, the de-identified data were made publicly available. The i2b2 organized 11 challenges, but datasets are only publicly available for 8 challenges.

The i2b2 started with the shared task of de-identification of clinical records, and identification of smoking status in 2006 (47,48). For the de-identification task, they provided 889 medical records to de-identify. They annotated all the records; 669 records were for training and 220 were for testing sets. The researchers annotated the 2006 i2b2 dataset with eight protected health information (PHI) categories including names of patients and doctors, hospitals, IDs, dates, locations, phone numbers, and ages. For the identification of smoking status, a subset (502) of 889 medical records was annotated and made available for the challenge.

The i2b2 obesity challenge dataset was developed in 2008 to automatically identify obesity and its fifteen comorbidities from clinical notes (49). These data have been used in several research studies about methods for the analysis of text data. For example, Lu et al. (2022) compared the model performance of seven deep learning methods for text classification (50). The researchers preprocessed the 2008 i2b2 dataset by removing numbers, punctuation symbols, and stop words, and by tokenization of the data. They noted that preprocessing reduced the size of the data. The authors found that the Transformer encoder deep learning method performed better than other deep learning methods for text classification. Su et al. (2021) examined several approaches for incorporating pre-trained sentence encoders into document-level representations of clinical text (51). They preprocessed the data using the removal of punctuation symbols and tokenization methods before applying it to the models. The researchers found that the best-performing approach was the transformer encoder-based model in identifying sixteen health conditions. Kumar et al. (2021) developed a classification system based on ensemble learning techniques over different combinations of classical machine learning and deep learning approaches to identify different health conditions using the 2008 i2b2 dataset (52). They compared the performance of classical machine learning and deep learning approaches in terms of different feature representations and feature selection methods. Since the preprocessing of text data has an impact on model performance, the researchers preprocessed the data using tokenization, removal of punctuation symbols and numeric values, and lemmatization. The results indicated that the ensemble learning technique slightly improved the performance of identification of health conditions compared to single classifier models.

Similarly, the 2014 i2b2 data has been used in many studies. These data contain information about diabetic patients with coronary artery disease (CAD) (53). The dataset was

used to develop methods for de-identification of UTD and to identify risk factors for CAD in diabetic patients over time. According to the US Health Insurance Portability Accountability Act (HIPAA), there are 18 categories of PHI; these include names, contact information, addresses, and ID numbers. The PHI in the 2014 i2b2 dataset was divided into seven main categories and 25 subcategories. The dataset was recently used to develop new text de-identification systems and to test and compare different de-identification systems. For example, Kocaman and Talby (2022) introduced a production-grade clinical and biomedical named entity recognition (NER) algorithm based on a modified bidirectional long short-term memory (Bi-LSTM)-convolutional neural network (CNN)-Char deep learning method into Apache Spark environment (54). The researchers trained NER models on eight different biomedical NER datasets and tested the performance of the model based on 3 different datasets, including the 2014 i2b2 dataset. They used sentence splitting and tokenization as preprocessing methods. The researchers found that the proposed algorithm performed comparatively better than commercial entity extraction tools (e.g., AWS Medical Comprehend and Google Cloud Healthcare API). Catelli et al. (2021) developed a new de-identification approach based on Bi-LSTM and Conditional Random Field (CRF) that captured latent syntactic and semantic similarities (55). The researchers corrected errors (i.e., inserting spaces when required and adjusting the start and end position of the PHI) and converted the XML files to brat standoff format and then to delimiter-separated values format in the preprocessing step. The preprocessed text was used as input for the NER system. This model was tested using the 2014 i2b2 dataset and model performance was similar to or higher than recent models with respect to category and binary recognition of PHI without any hand-written rules.

The latest dataset available from the i2b2 challenges is the 2018 i2b2 clinical challenge dataset. There were two shared tasks: cohort selection for clinical trials and adverse drug events and medication extraction in EMRs (56,57). For the cohort selection task, the dataset was a subset of the 2014 i2b2 dataset. The organizers annotated the data whether a patient met certain selection criteria for clinical trials. The dataset for the adverse drug events and medication extraction was collected from the MIMIC III medical records. The researchers annotated 505 clinical notes and made them available for the challenge.

MIMIC is a large, and publicly available database containing de-identified medical records of patients admitted at the BIDMC (58). There are three versions of MIMIC datasets available over the years: MIMIC-IV, MIMIC-III, and MIMIC-II. MIMIC III is a widely used dataset for UTD research among the MIMIC datasets. MIMIC-III data contains information for more than forty thousand patients who were admitted to critical care units of the BIDMC between 2001 and 2012. The dataset has different kinds of information including demographics, vital signs, laboratory test results, procedures, medications, caregiver notes, imaging reports, and mortality (both in and out of hospital). However, these data have not been annotated, which limits their usability for predictive modeling and de-identification.

2.5 Summary

Previous studies used different preprocessing methods and different combinations of methods to prepare unstructured text data for research. These studies reveal that text data preprocessing increased the quality of the data and/or performance of models that were subsequently applied to text data. Only a few studies focused on medical text data. However, previous research has not examined the effect of the order and number of preprocessing methods on different types of analyses applied to UTD in EMR data.

The i2b2 datasets were used for different purposes; the 2008 i2b2 dataset was used to detect obesity and fifteen comorbidities, the 2014 i2b2 dataset was used to develop automated de-identification methods, and the 2018 i2b2 dataset was used to select cohorts based on their clinical and demographic characteristics. The i2b2 datasets contain high-quality annotations for each specific task and are publicly available, which makes them an optimal choice for researchers to use. Researchers have used these datasets to develop new models or to compare the performance of existing machine learning or deep learning models. Though several studies used preprocessing methods to prepare the i2b2 datasets for specific tasks, no studies using these datasets have examined the effect of the number or order of preprocessing methods on task performance.

Chapter 3: Methods

In this chapter, I describe the study design, data sources, and study variables. The latter includes the preprocessing methods and the performance measures used to evaluate the machine learning models for UTD. The data extraction technique and model development for each of the objectives with statistical analyses are discussed.

An ethics approval waiver was sought from the University of Manitoba Health Research Ethics Board because the study used publicly available data from the i2b2 clinical data challenges. The waiver was accepted by the Department of Community Health Sciences, Max Rady College of Medicine, University of Manitoba.

3.1 Study design to test research hypotheses

An experimental design with two factors was used to test the study hypotheses. The order of preprocessing methods was independent and was nested within the number of preprocessing methods. Consequently, a nested design was adopted. There were two levels for the number of methods: four and five. There were two orders of methods with four preprocessing methods and six orders of methods with five preprocessing methods. The number and order of preprocessing methods were treated as random and fixed factors, respectively.

3.2 Data sources

Three i2b2 datasets were used for this research. As noted previously, these datasets were adopted for data challenges about methods to analyze clinical records, including methods for UTD. A key reason for selecting these datasets is that they were annotated by medical experts. The datasets were widely used in prior research because of their availability and annotation for specific tasks. For example, the 2014 i2b2 dataset has been used to develop new text data de-identification approaches (54,55,59–62).

For Objective 1, to assess the effect of order and number of preprocessing methods on detecting different health conditions in text data, the 2008 i2b2 obesity challenge dataset was used. The dataset contains 1237 health records for patients hospitalized for obesity or diabetes (49). Two obesity experts annotated each record, to identify obesity and 15 comorbidities associated with obesity. The comorbidities included asthma, atherosclerotic cardiovascular disease (CAD), congestive heart failure (CHF), depression, diabetes mellitus (DM), gallstones/cholecystectomy, gastroesophageal reflux disease (GERD), gout, hypercholesterolemia, hypertension, hypertriglyceridemia, obstructive sleep apnea (OSA), osteoarthritis (OA), peripheral vascular disease (PVD), and venous insufficiency. The experts first classified each disease as present, absent, questionable, or unmentioned based on explicitly documented information in the health records; these annotations are known as textual judgments. The experts also performed intuitive judgments, which were based on both explicitly documented information and contextual information in the health records. The experts classified each disease as present, absent, or questionable based on all information in the health records. The 2008 i2b2 obesity challenge focused on the development of models that automatically extract information about the presence of obesity and the 15 comorbidities. Model performance in the challenge was measured by precision, recall, and F1 score. I used the 2008 i2b2 dataset to identify obesity and 15 comorbidities using intuitive judgments as the reference standard. There were no intuitive judgments for the two health records, so they were removed from my analysis. Accordingly, my analysis is based on data contained in 1235 health records. There were very few observations (0.2%) in the questionable class compared to other classes which might affect model performance. I therefore merged the absent and questionable classes.

The 2018 i2b2 challenge dataset was used to assess the effect of order and number of preprocessing methods on identifying cohort selection criteria and measuring the similarity of information for Objectives 2 and 3, respectively. The dataset contains 1267 clinical notes from 288 patients, with each patient having two to five notes (56). Thirteen cohort selection criteria collected from real-world studies in ClinicalTrials.gov were adopted for these data. The cohort selection criteria represent demographic and clinical patient characteristics, including present medical conditions, past medical conditions, current treatments and reasons for treatments, lab test results, use of over-the-counter medications, abuse of drugs/alcohol, and language spoken. The cohort selection criteria are described in Table 3.1. Two medical experts annotated the records for each patient as meeting or not meeting each of the selection criteria.

Table 3.1: Description of the cohort selection criteria annotated in the 2018 i2b2 dataset.

Criterion	**Description**
Abdominal	History of intra-abdominal surgery, small or large intestine resection, or small bowel obstruction
Advanced-CAD	Presence of advanced cardiovascular disease
Alcohol-abuse	Current alcohol use exceeds the weekly recommended limits
ASP-for-MI	Using aspirin to prevent myocardial infarction (MI)
Creatinine	Serum creatinine is larger than the upper limit of normal
DIETSUPP-2MOS	Dietary supplements used (excluding vitamin D) in the last 2 months
Drug-abuse	Current or past drug abuse
English	English spoken patient
HBA1C	Any hemoglobin A1c (HbA1c) value is between 6.5% and 9.5%
KETO-1YR	Diagnosis of ketoacidosis in the last year
Major-diabetes	Diabetes-related major complication
Makes-decisions	Medical decisions must be taken by patients
MI-6MOS	MI in the last 6 months

Note: CAD = cardiovascular disease, ASP = aspirin, MI = myocardial infarction, DIETSUPP = dietary supplements, HBA1C = hemoglobin A1c, KETO = ketoacidosis, YR = year, MOS = months

For Objective 3, to assess the effect of order and number of preprocessing methods on measuring the similarity of information for the same individual, a pair of clinical notes was selected for each patient from the 2018 i2b2 dataset. This resulted in 576 clinical notes for 288 patients. The notes for each patient did not necessarily come from the same physician.

The 2014 i2b2/UTHealth challenge dataset was used for Objective 4, to assess the effect of order and number of preprocessing methods on de-identification of UTD. The dataset contains 1304 health records for 296 patients (64). The dataset provides information over time about diabetic patients with coronary artery disease, their family backgrounds, and medical histories. The original challenge had four tasks; the first task was the de-identification of UTD (64). Protected health information (PHI) in the health records was assigned to seven main categories and 25 subcategories for the original challenge. The PHI main categories include name (subcategories: patient, doctor, username), profession, location (subcategories: hospital, organization, street, city, state, country, zip, other), age, date, contact (subcategories: phone, fax, email, URL, IP address), and IDs (subcategories: social security number, medical record number, health plan number, account number, license number, vehicle ID, device ID, biometric ID, ID number). The challenge organizers annotated the data for each type of PHI. Precision, recall, and F1 score were used to evaluate model performance for PHI in the original challenge.

3.3 Study variables

The independent variables in the experimental design used to test the research hypotheses were the order of preprocessing methods and the number of preprocessing methods. The dependent variables in the experimental design used to test the research hypotheses were measures of model performance and text similarity.

3.3.1 Independent variables

The order and number of methods to preprocess the datasets for the study objectives are described in Table 3.2. With respect to the order of methods, tokenization was always the first method, and word stemming or lemmatization was always the last method in every order (63–71). Since tokenization identifies words that create a string of characters and a text is understandable by its words, tokenization was implemented first. Word stemming and lemmatization are similar preprocessing methods because they both create root words or base words. Accordingly, both preprocessing methods were included in the study.

Table 3.2: Order and number of preprocessing methods for the study objectives.

	Number of preprocessing methods		
Order	**4**	**5 (with word stemming)**	**5 (with lemmatization)**
1	T-RPS-CSE-WS	T-RPS-EA-CSE-WS	T-RPS-EA-CSE-L
2	T-CSE-RPS-WS	T-RPS-CSE-EA-WS	T-RPS-CSE-EA-L
3	-	T-CSE-RPS-EA-WS	T-CSE-RSP-EA-L
4	-	T-CSE-EA-RPS-WS	T-CSE-EA-RPS-L
5	-	T-EA-RPS-CSE-WS	T-EA-RPS-CSE-L
6	-	T-EA-CSE-RPS-WS	T-EA-CSE-RPS-L

Note: T = Tokenization, RPS = Removal of punctuation symbols, CSE = Correction of spelling errors, EA = Expansion of abbreviations, WS = Word stemming, L = Lemmatization

The preprocessing methods were selected based on previous research that used these i2b2 datasets. For the 2008 i2b2 challenge, several teams reported that they preprocessed the UTD prior to the classification task (Table A1 in the Appendix), including sentence splitting, POS tagging, shallow parsing, expansion of abbreviations, and tokenization. For the 2018 i2b2 data, the preprocessing methods mentioned by some of the teams included sentence splitting, lowercase conversion, tokenization, removal of stop words and punctuation symbols, spelling correction, and lemmatization (Table A1 in the Appendix). Similar methods were also used by some of the teams that analyzed the 2014 i2b2 dataset. As well, removal of punctuation symbols,

word stemming, lemmatization, and correction of spelling errors are commonly used preprocessing methods for EMR data (72). Therefore, the study used tokenization, removal of punctuation symbols, correction of spelling errors, expansion of abbreviations, word stemming, and lemmatization to preprocess the data for all objectives.

Removal of stop words was initially selected as a preprocessing method. However, I did not retain this preprocessing method after conducting some preliminary analyses of the effect of stop word removal on detection of health conditions, identifying cohort selection criteria, and measuring the similarity of information for Objectives 1, 2, and 3, respectively (see Tables A2-A4 in the Appendix). Stop words capture structural information about text data, and this can affect model performance (21). For example, studies that have conducted classification tasks using text data found that the presence of stop words resulted in better performance than when stop words were removed (73–75).

Some information (e.g., individual names, dates, phone/contact numbers, batch numbers, and documentation code) was removed from both the 2008 and 2018 i2b2 datasets before applying the preprocessing methods. This step was taken because these types of data were detected as incorrect spellings. In addition, these were not related to the health conditions or cohort selection criteria for the 2008 and 2018 i2b2 datasets, respectively (76–78). Removal of this information had no impact on model results in preliminary assessments on the detection of health conditions, identification of cohort selection criteria, and similarity measures for Objectives 1, 2, and 3, respectively. (see Tables A5-A7 in the Appendix).

The preprocessing methods adopted in my study were:

Tokenization: This is a method of dividing texts into words, phrases, or other meaningful parts known as tokens (19,64). There are several libraries in python that can be used to tokenize text

data. The study used the Natural Language Toolkit (NLTK) library of python since it is a popular and commonly used python library for tokenization.

Removal of punctuation symbols: Punctuation symbols are redundant in text data analysis and it is common to remove them (18). There are a few python libraries for punctuation symbol removal; these symbols can also be detected and removed using regular expressions. The python library, named string, was used to remove punctuation symbols as well as special characters (e.g., <, >, #, *, |, (,), and @). I wrote a function to remove the punctuation symbols before and after a token, but not within a token. For example, "94.3" was unchanged, but "96%" was changed to "96" after the removal of the punctuation symbol.

Expansion of abbreviations: Expansion of abbreviations helps in understanding medical text and can improve the accuracy of information extraction (79). I used a list of abbreviations that included more than three thousand medical abbreviations (80). This abbreviation dictionary was developed for a python library, MEDIALpy, which was used to check medical abbreviations from the text. A function was written to look up the abbreviations from the list of tokens and replace them with expansions. There were many abbreviations that had more than one expansion in the abbreviation list. To automate the process, I randomly chose one of the expansions when there was more than one expansion for an abbreviation in the abbreviation dictionary. Since the expansion might be a phrase (more than one word), I tokenized the expansion after replacing the abbreviation.

Correction of spelling errors: Spelling errors often occur in medical records; these errors might lead to misinterpretation of the text data and de-identification errors. I used the pyspellchecker library of python that uses the Levenshtein Distance (difference between two strings) algorithms to correct spelling errors. It is one of the simplest spell-checker algorithms that provides the best

corrected word in place of the misspelled word. I attached a dictionary that contains medical terms along with regular English words by using the spell checker library. This medical term dictionary was based on two published medical dictionary projects (81).

Word stemming: Stemming is a process of removing the endings of words in order to obtain their root form or stem. For example, presents, presenting, and presented can be truncated to the stem present. Though there were stemming options in the NLTK library of python; I used the snowball stemmer because it is more current than the porter stemmer and is an efficient tool (82).

Lemmatization: The process of reducing a word to its canonical or dictionary word is known as lemma. Lemmatization links similar meaningful words to a common meaningful word. Lemmatization provides a meaningful word, whereas the root word resulting from stemming may not have meaning or correct spelling. For example, the lemmatization of study, studying, and studies would return the base word study, but the stemming of study, studying, studies, or studied would return the root studi. However, stemming is faster than lemmatization since lemmatization requires look-up tables to provide meaningful words. I used the wordnet lemmatizer from the NLTK library of python which is a common tool for lemmatization.

3.3.2 Dependent variables

There were several dependent variables investigated in this study, including model performance measures and similarity measures. Model performance was evaluated for Objectives 1, 2, and 4 using sensitivity, specificity, F1 score, accuracy, and precision. Sensitivity refers to the proportion of all true positive values correctly identified as positive by the model. Specificity refers to the proportion of all true negative values correctly identified as negative. The F1 score is the harmonic mean of precision and sensitivity. Accuracy is the proportion of correctly predicted observations in the true class. Precision is the proportion of predicted positive

value correctly identified as positive (83). Sensitivity, specificity, F1 score, accuracy, and precision can be defined using a confusion matrix (Table 3.3) that delineates whether a model predicts the true class correctly.

Table 3.3: Confusion matrix.

	True class	
Predicted class	Positive	Negative
Positive	True Positive (TP)	False Positive (FP)
Negative	False Negative (FN)	True Negative (TN)

The definitions of the performance measures based on this confusion matrix are:

$$Sensitivity = TP/(TP + FN) \quad 3.1$$

$$Specificity = TN/(TN + FP) \quad 3.2$$

$$F1\ score = 2TP/(2TP + FP + FN) \quad 3.3$$

$$Accuracy = (TP + TN)/(TP + TN + FP + FN) \quad 3.4$$

$$Precision = TP/(TP + FP) \quad 3.5$$

For Objective 1, to assess the effect of preprocessing methods on detecting obesity and 15 comorbid conditions, sensitivity, specificity, F1 score, accuracy, and precision were estimated for each of these 16 health conditions. Mean values of sensitivity, specificity, F1 score, accuracy, and precision were calculated across all health conditions, along with their standard deviations. This approach is consistent with the original challenge evaluation method. The mean values were used to test for differences among the orders and number of methods.

For Objective 2, to assess the effect of preprocessing methods on identifying cohort selection criteria, I first estimated sensitivity, specificity, F1 score, accuracy, and precision for each criterion. Mean values of sensitivity, specificity. F1 score, accuracy, and precision were

calculated across all criteria, along with their standard deviations. These mean values were used to test the research hypotheses.

For Objective 3, to assess the effect of preprocessing methods on measuring the similarity of information, cosine similarity was estimated for pairs of notes from the 2018 i2b2 data. Cosine similarity is widely used to measure the similarity between documents; it does not depend on the size of the document (i.e., the number of words) (84). Cosine similarity is the inner product of two vectors (85). If $\boldsymbol{A}$ and $\boldsymbol{B}$ are two vectors that are obtained by transforming the features of documents into numerical vectors, cosine similarity is calculated as:

$$similarity(A, B) = \cos(\theta) = \frac{A.B}{||A||||B||} \qquad 3.6$$

In equation 3.6, cos(θ) is the cosine of the angle between two vectors. The numerator is the dot product of the vectors, and the denominator is the product of the Euclidean norms for the vectors. The value of cosine similarity lies between 0 and 1. A value close to one indicates high similarity, while a value close to zero indicates high dissimilarity. I calculated cosine similarity for each pair of notes. Mean cosine similarity was calculated for all pairs of notes, along with its standard deviation.

Sensitivity, specificity, F1 score, accuracy, and precision were used to evaluate model performance for Objective 4, to assess the effect of preprocessing on de-identification of UTD (53). Evaluation metrics were first estimated for each document at the entity level (i.e., matching the exact beginning and end positions of each PHI, as well as tag name and type attribute). Mean sensitivity, specificity. F1 score, accuracy, and precision were calculated across all documents, along with their standard deviations. The mean values were used to test for differences among the order and number of methods.

3.4 Text data extraction and model development

For Objectives 1 and 2 (classification tasks), and Objective 3 (similarity measures), text data were extracted using TF-IDF to transform the text data into numerical vectors that were subsequently analyzed using machine learning models. For Objectives 1 and 2, machine learning models were used to detect health conditions and cohort selection criteria, respectively. For Objective 3, text similarity was assessed by cosine similarity. For Objective 4, a Bi-LSTM-CRF model-based deep learning model was used to de-identify the UTD. Figure 3.1 shows a diagram of the research process.

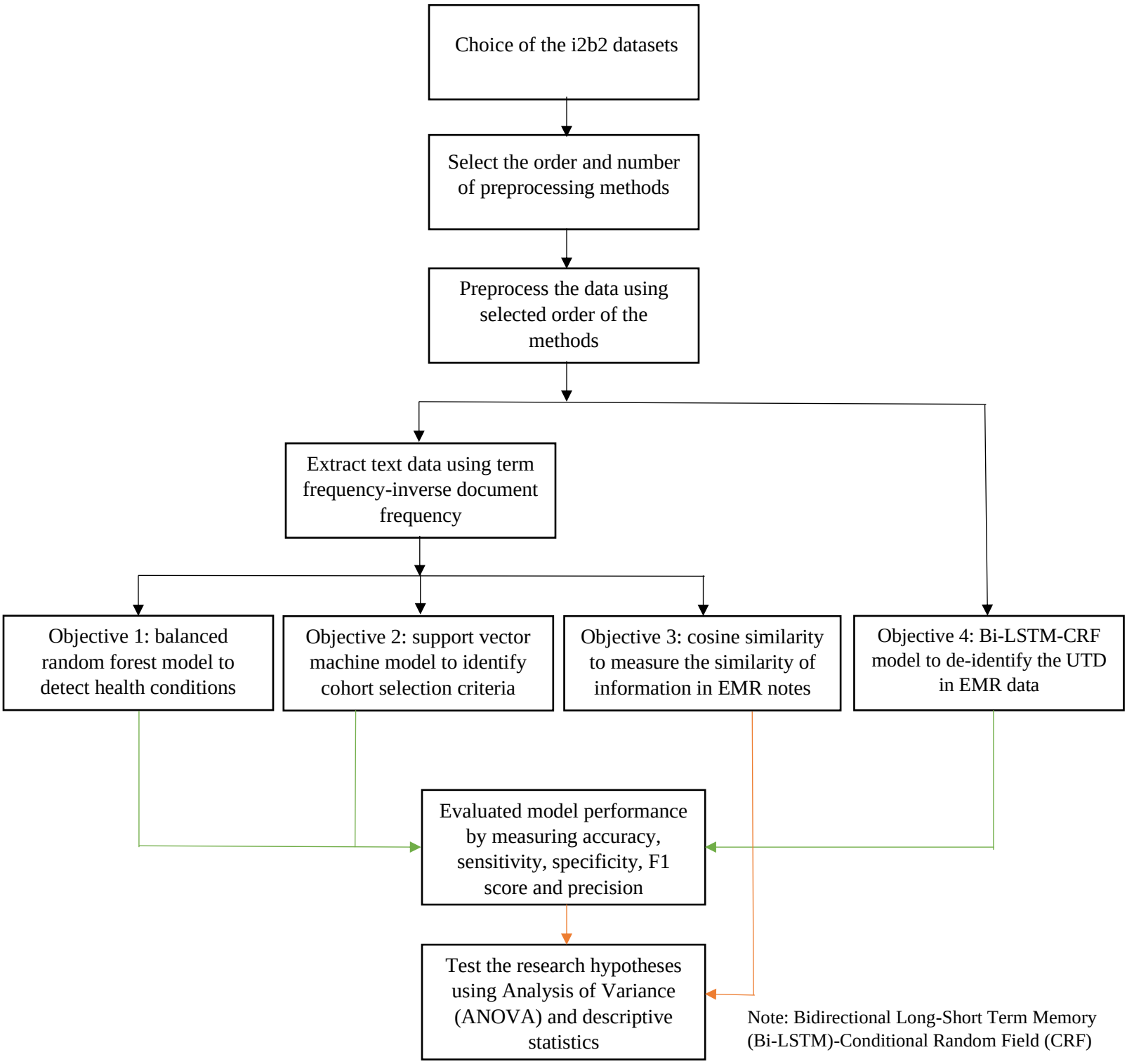


Figure 3.1: Flowchart of the research process.

3.4.1 Text data extraction

When using text data for analysis, the data first needs to be transformed into a numerical vector since machine learning models often use numerical data and this process is known as feature extraction. There are a few approaches (e.g., bag-of-words, TF-IDF, and word2vec) that can be used to accomplish this transformation. For classification tasks and similarity measures, this study used the TF-IDF approach, a widely used measure in NLP and information retrieval because of its simplicity in implementation. While some studies define TF-IDF as a preprocessing method (3,12,13), other studies define TF-IDF as a separate step in feature extraction or document modeling (64,86,87). In this study, the preprocessed data were transformed into numerical vectors to use for classification tasks and similarity measures. However, TF-IDF was not used for Objective 4, in the de-identification tasks.

TF-IDF provides information about the importance of a word to a document in a collection of documents. It is obtained by multiplying term frequency (TF) and inverse term frequency (IDF). TF refers to how often a word appears in a document, and IDF refers to how often a word appears in a collection of documents. The TF-IDF vectorizer from the scikit-learn library of python was used to transform the text into vectors in this study.

3.4.2 Objective 1: Model development to detect health conditions in UTD

For Objective 1, the effect of preprocessing methods on the correct detection of health conditions in EMR data was investigated. Teams that used the study data in prior data challenges employed rule-based, machine learning, or a combination of rule-based and machine learning approaches to detect health conditions (49). One of the rule-based approaches performed the best; however, this approach requires pre-specified rules that can be difficult and time-consuming to develop. While none of the teams adopted a random forest (RF) model to detect

health conditions, a number of studies have shown that the RF method results in better performance for text classification tasks than other machine learning methods (14,15,67,86,88–93). Moreover, to address class imbalance this study used the balanced random forest (BRF) model, which is an extension of the RF model. When there is a severe imbalance in the data, the RF model often results in poor performance for the minority class, while the BRF model, which uses under sampling, works better (Table A8 in the Appendix). The BRF model is an ensemble learning algorithm in which decision trees are chosen so that minority classes are given greater weight. In an RF model, each decision tree is built randomly and independently, but in the BRF model decision trees are created using a balanced subset of the data. For each iteration in the model, a bootstrap sample of the minority class is drawn, and the same number of observations are randomly drawn from the majority class with replacement. This approach ensures that the decision trees are not biased to the majority class and have a better representation of the underlying patterns of the data. The number of features used in bootstrap sampling was equivalent to the square root of all features. This process helps to reduce the variance and de-correlates the trees (94). This process continues for a pre-specified number of iterations. Then the predictions are aggregated, and a final prediction is produced.

Five-fold cross validation was used to split the data into training and test sets. A total of 1000 trees were used in the models. The number of features was the square root of the total number of features. Default values were selected for other hyperparameters. The models were fitted to the data using version 0.11 of the imbalanced learn library of python. For example, the "auto" sampling strategy was used; it resamples all classes except the minority class.

3.4.3 Objective 2: Model development to detect cohort selection criteria in UTD

The second objective was to assess the effect of the number and order of preprocessing methods on cohort selection criteria for clinical trials from UTD. Most of the top-ranked teams in the 2018 i2b2 challenge used either rule-based or hybrid (i.e., a combination of rule-based and machine learning-based methods) methods. A support vector machine (SVM) was also used to identify cohort selection criteria and it performed better as an individual approach compared to other machine learning or deep learning methods in prior research (95). Other studies have also shown that an SVM performs better than other machine learning classifiers (86,87,90,96). Accordingly, an SVM was adopted for the current study. The SVM model is a supervised machine learning model (97) that identifies a hyperplane in high-dimensional space that can distinctly classify the data points. The main advantages of using an SVM model are: the model works very fast and provides better performance with smaller sample sizes than other classifiers. An SVM model can be used for both linear and non-linear classifications.

In this study, I used a 5-fold stratified cross validation technique for the SVM model. Since there were class imbalances in the data, this technique worked better compared to other k-fold cross validation. A linear SVM model was used that aligned with previous research using these data (95). To choose the hyperparameters for the model, I performed a grid search for some of the primary hyperparameters such as regularization parameter and class weight. I found that the default values of this hyperparameter (i.e., 1.0) with class weight “balanced” provided better performance than other options. Other hyperparameters were set at default values aligned with prior research (95). For example, the maximum number of iterations was 1000 and the penalty was l2. The sci-kit learn library of python, version 1.0.2, was used to train the linear SVM model.

There was only one observation in the “met” class for the criterion of diagnosis of ketoacidosis in the past year. Training of the SVM model requires observations in both classes. When using a 5-fold stratified cross validation technique, data were randomly split into train and test and sometimes there was no observation for the “met” class. To overcome this problem, I used random over sampling for this criterion to increase the sample size in the “met” class. The initial results of keeping the only observation in the training data were similar to the results of using the oversampling technique. Additionally, I considered “met” and “not met” as “present” and “absent” outcomes, respectively.

3.4.4 Objective 3: Model development to assess document similarity

The third objective was to assess the effect of preprocessing methods on the similarity of information contained in EMR notes. The presence of similar notes in EMR data is potentially problematic in training predictive models (98). For example, similar notes may skew the statistics of term frequencies and may cause overfitting of the trained model. A poor predictive model may incorrectly identify correlations between symptoms and health conditions. Similar or duplicate notes increase the size of text datasets, which may affect the computational resources required to analyze them. Removing very similar notes from a dataset may reduce computational time by reducing the total dataset size. Accordingly, measuring the similarity of text information could be beneficial as an initial step in model development.

I used the TF-IDF vectorizer to transform the preprocessed data into numerical vectors. I estimated cosine similarities using the corresponding function from the scikit-learn library of python.

3.4.5 Objective 4: Model development to de-identify UTD

The fourth objective was to assess the effect of the number and order of preprocessing methods on accurate de-identification of UTD in EMR data. The dataset I used for this objective was from the 2014 i2b2 challenge, in which ten teams developed systems for the de-identification of UTD. Most of the teams used the CRF with hand-written rules (53). None of the teams used deep learning methods for de-identification, but subsequent research has demonstrated that deep learning-based systems perform better than the CRF model. However, artificial neural networks (ANNs) do not require pre-specified rules or features. ANNs can automatically generate effective features by performing composition over tokens. I used an ANN model, named the Bi-LSTM-CRF model, which is a supervised neural method and a promising approach for de-identification (99–103). Huang et al. (104) first introduced the Bi-LSTM-CRF model for sequence tagging to de-identify PHI in EMRs. A Bi-LSTM consists of two LSTMs; one LSTM calculates the forward hidden states, and the other one calculates the backward hidden states. The Bi-LSTM-CRF model uses past and future features through the Bi-LSTM layer and sentence-level tag information through the CRF layer.

This study used the de-identification system based on the Bi-LSTM-CRF model that was developed by Trienes et al. (99). Pre-trained GloVe embedding was used in the embedding layer on a word level with contextual Flair embeddings. This study used the same hyperparameters that were used in the original study. For example, the initial learning rate was 0.1 and it was halved when the training loss did not decrease for 3 consecutive epochs and stopped if the learning rate fell below 0.0001 or if 150 epochs were reached. The number of hidden layers in the LSTM was set to 1 with 256 recurrent units. The model used a locked dropout of 0.5 and a

mini-batch size of 32. The dataset was split into training, validation, and testing sets with a 60/20/20 ratio.

3.5 Statistical analyses

Characteristics of the datasets, including tokens, punctuation symbols, spelling errors, and abbreviations were described using frequencies, means, standard deviations, medians, and inter-quantile ranges. Flow charts were used to illustrate the number of tokens that were removed from each dataset after each preprocessing method. The outcome variables associated with each objective, including sensitivity, specificity, F1 score, accuracy, precision, and cosine similarity, were descriptively analyzed using means and standard deviations.

To test the research hypotheses, given that the order of the methods was nested within the number of methods, it was not possible to test the interaction of both independent variables. Therefore, I used a one-way analysis of variance (ANOVA) to test the effect of the order of methods; this analysis was conducted separately at each level of number of methods. The assumptions of the ANOVA were assessed using visual methods (e.g., histogram) and descriptive statistics, including the mean (standard deviation) and median (interquartile range), as well as measures of distribution shape (i.e., skewness, kurtosis) for each outcome variable. Since inferential methods to test the assumption of a normal distribution are sensitive to sample size, the study used skewness and kurtosis to descriptively assess the assumption of normality. An absolute skewness value > 2 or an absolute kurtosis > 7 were used as rules of thumb to descriptively assess departures from a normal distribution, where a normal distribution has a skewness of 0 and of kurtosis of 3 (105,106). The differences among the order of methods for each level of the number of methods were tested using a nominal $\alpha = 0.05$. Differences in outcome variable values for four and five preprocessing methods were descriptively analyzed

using means and standard deviations. All descriptive analyses were conducted using python (version 3.9). The ANOVA was performed with a built-in function in R (version 4.2.3).

Chapter 4: Results for classification of text-based measures

4.1 Dataset characteristics

Table 4.1 shows that there were a similar number of clinical notes in both datasets that were used for classification tasks. The 2008 i2b2 dataset had a mean of 1266.0 (standard deviation [SD] 554.4) tokens per note, which was 1.5 times higher than the mean tokens per note in the 2018 i2b2 dataset. The 2008 i2b2 dataset had a mean (SD) of 242.7 (116.2) punctuation symbols, 81.4 (47.4) spelling errors, and 126.8 (66.3) abbreviations per note. The 2018 i2b2 dataset had a mean (SD) of 146.8 (100.1) punctuation symbols, 54.7 (46.1) spelling errors, and 92.3 (64.4) abbreviations per note.

Table 4.1: Characteristics of the 2008 and 2018 i2b2 datasets.

Attributes	2008 i2b2 dataset			2018 i2b2 dataset		
	Frequency	Mean (SD) per note	Median (Q1, Q3) per note	Frequency	Mean (SD) per note	Median (Q1, Q3) per note
Clinical notes	1235	-	-	1267	-	-
Tokens	1563483	1266.0 (554.4)	1178.0 (889.0, 1561.5)	946847	747.3 (432.0)	634.0 (434.0, 984.0)
Punctuation symbols	299697	242.7 (116.2)	224.0 (158.0, 300.0)	186000	146.8 (100.1)	119.0 (84.0, 181.5)
Spelling errors	100497	81.4 (47.4)	72.0 (49.0, 102.5)	69293	54.7 (46.1)	40.0 (20.0, 77.0)
Abbreviations	156561	126.8 (66.3)	117.0 (78.0, 163.0)	116990	92.3 (64.4)	73.0 (44.0, 126.0)

Note: SD = Standard deviation; Q1 = 1st quartile, Q3 = 3rd quartile

For Objective 1, the 2008 i2b2 dataset had 1235 clinical notes; there was a single note for each patient. The dataset was annotated to identify 16 health conditions classified as present and absent; the frequencies of these conditions are described in Table 4.2. When the annotators disagreed in identifying any of the health conditions for a patient, the patient was not considered for annotation of that specific health condition. For example, the first patient was annotated for

10 of 16 health conditions. There was also substantial class imbalance for some of the health conditions, such as hypertriglyceridemia, which had a prevalence of only 5.2%.

Table 4.2: Frequency of health conditions in the 2008 i2b2 dataset.

Health Conditions	**Present (%)**	**Absent (%)**	**Total**
Asthma	154 (13.4)	999 (86.6)	1153
CAD	663 (59.2)	456 (40.8)	1119
CHF	513 (48.2)	552 (51.8)	1065
Depression	247 (21.0)	927 (79.0)	1174
Diabetes	806 (69.4)	356 (30.6)	1162
Gallstones	181 (15.1)	1020 (84.9)	1201
GERD	237 (23.3)	781 (76.7)	1018
Gout	155 (12.8)	1057 (87.2)	1212
Hypercholesterolemia	557 (53.9)	477 (46.1)	1034
Hypertension	869 (80.2)	215 (19.8)	1084
Hypertriglyceridemia	62 (5.2)	1126 (94.8)	1188
OA	208 (18.3)	926 (81.7)	1134
Obesity	477 (42.9)	635 (57.1)	1112
OSA	165 (13.7)	1043 (86.3)	1208
PVD	175 (15.5)	957 (84.5)	1132
Venous Insufficiency	83 (7.8)	975 (92.2)	1058

Note: CAD = atherosclerotic cardiovascular disease, CHF = congestive heart failure, GERD = gastroesophageal reflux disease, OSA = obstructive sleep apnea, OA = osteoarthritis, PVD = peripheral vascular disease

The 2018 i2b2 dataset was used for Objective 2 and the dataset was annotated to identify 13 cohort selection criteria. There were 1267 clinical notes for 288 patients and at least two notes per patient. Each of the patients was annotated for all the criteria. Hence, the total number of patients for each criterion was 288. Table 4.3 shows that half of the criteria was severely imbalanced (i.e., minority class had less than 10% of observations).

Table 4.3: Frequency of 13 cohort selection criteria in the 2018 i2b2 dataset.

Criterion	Present (%)	Absent (%)
Abdominal	107 (62.8)	181 (37.2)
Advanced-CAD	170 (59.0)	118 (41.0)
Alcohol-abuse	10 (3.5)	278 (96.5)
ASP-for-MI	230 (79.9)	58 (20.1)
Creatinine	106 (36.8)	182 (63.2)
DIETSUPP-2MOS	149 (51.7)	139 (48.3)
Drug-abuse	15 (5.2)	273 (94.8)
English	265 (92.0)	23 (8.0)
HBA1C	102 (35.4)	186 (64.6)
KETO-1YR	1 (0.3)	287 (99.7)
Major-diabetes	156 (54.2)	132 (45.8)
Makes-decisions	277 (96.2)	11 (3.8)
MI-6MOS	26 (9.0)	262 (91.0)

Note: CAD = cardiovascular disease, ASP = aspirin, MI = myocardial infarction, DIETSUPP = dietary supplements, HBA1C = hemoglobin A1c, KETO = ketoacidosis, YR = year, MOS = months

4.2 Effect of preprocessing methods

Figure 4.1 shows that there were 1.48 million tokens at the beginning of preprocessing; this number was reduced to 1.24 million tokens after using four preprocessing methods for order 1 in the 2008 i2b2 dataset. There were 18.5% of tokens with punctuation symbols that were removed during preprocessing. The most common punctuation symbols were sentence-ending periods, commas, colons, and round brackets. There were 5.3% of tokens with incorrect spelling that were corrected during preprocessing. When I used word stemming to reduce the tokens to their stems, 23.0% of tokens were stemmed.

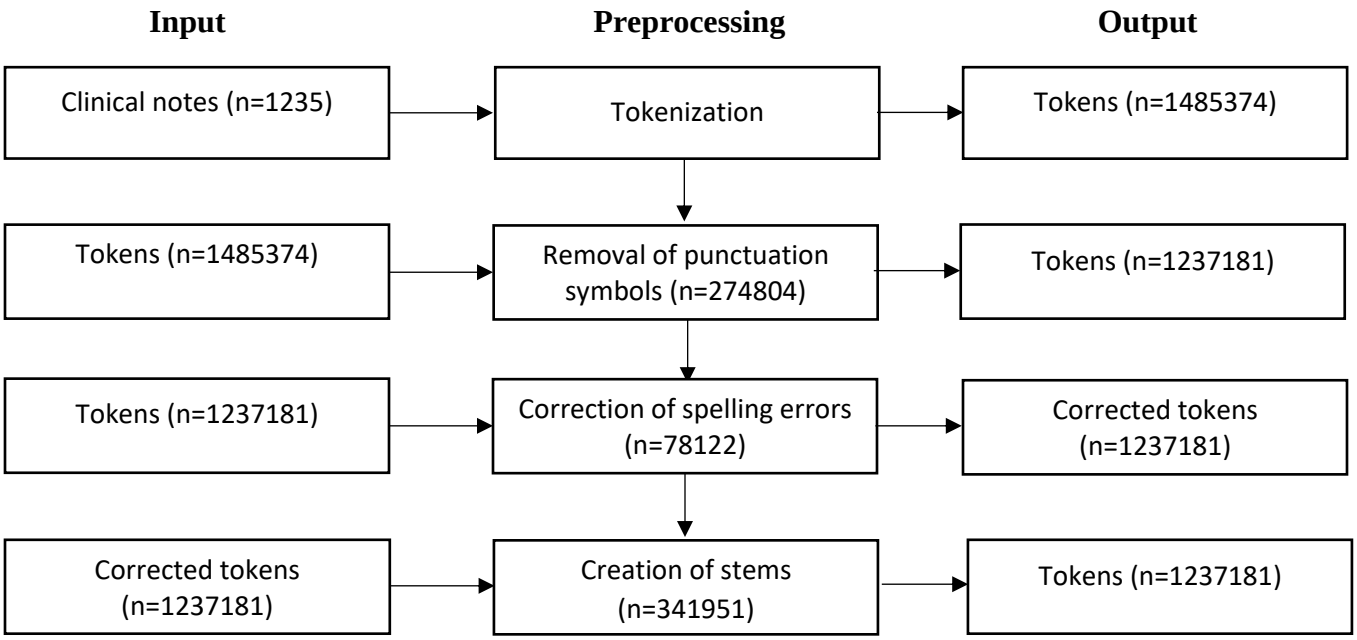


Figure 4.1: Effect of preprocessing methods in the 2008 i2b2 dataset using four preprocessing methods for order 1 (See Table 3.2 for the order of preprocessing methods).

Figures 4.2 and 4.3 show the effect of preprocessing methods for the 2008 i2b2 dataset when five preprocessing methods were applied to the data. Since I expanded abbreviations, the number of tokens was 1.4 million after preprocessing. The percentages of punctuation symbols and spelling errors were similar when four or five preprocessing methods were used. There were 10.1% of tokens with abbreviations. A large number of tokens (33.6%) were stemmed (Figure 4.2), whereas only 5.7% of tokens were lemmatized (Figure 4.3). This might happen because stemming uses heuristic rules, but lemmatization uses linguistic rules, and it provides the canonical form of a word.

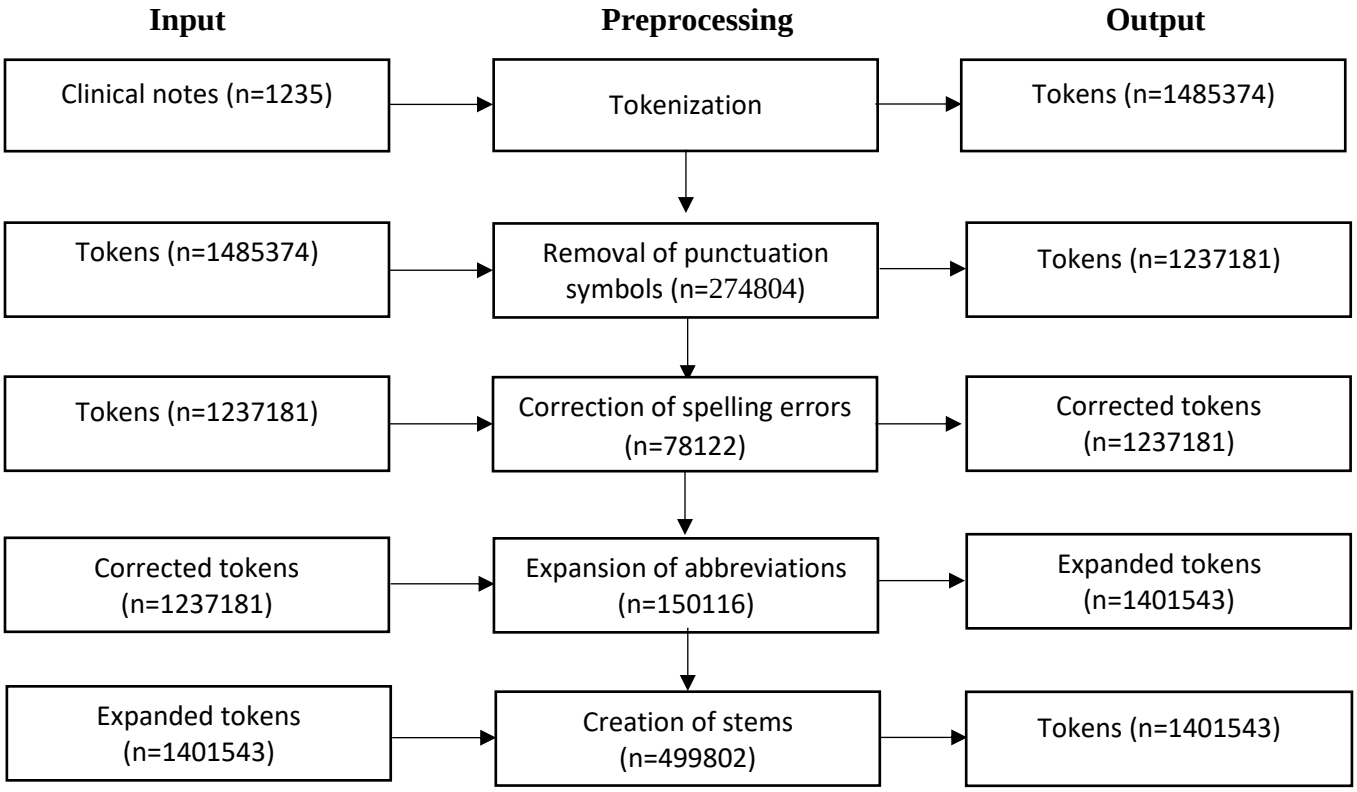


Figure 4.2: Effect of preprocessing methods in the 2008 i2b2 dataset using five preprocessing methods including stemming for order 1 (See Table 3.2 for the order of preprocessing methods).

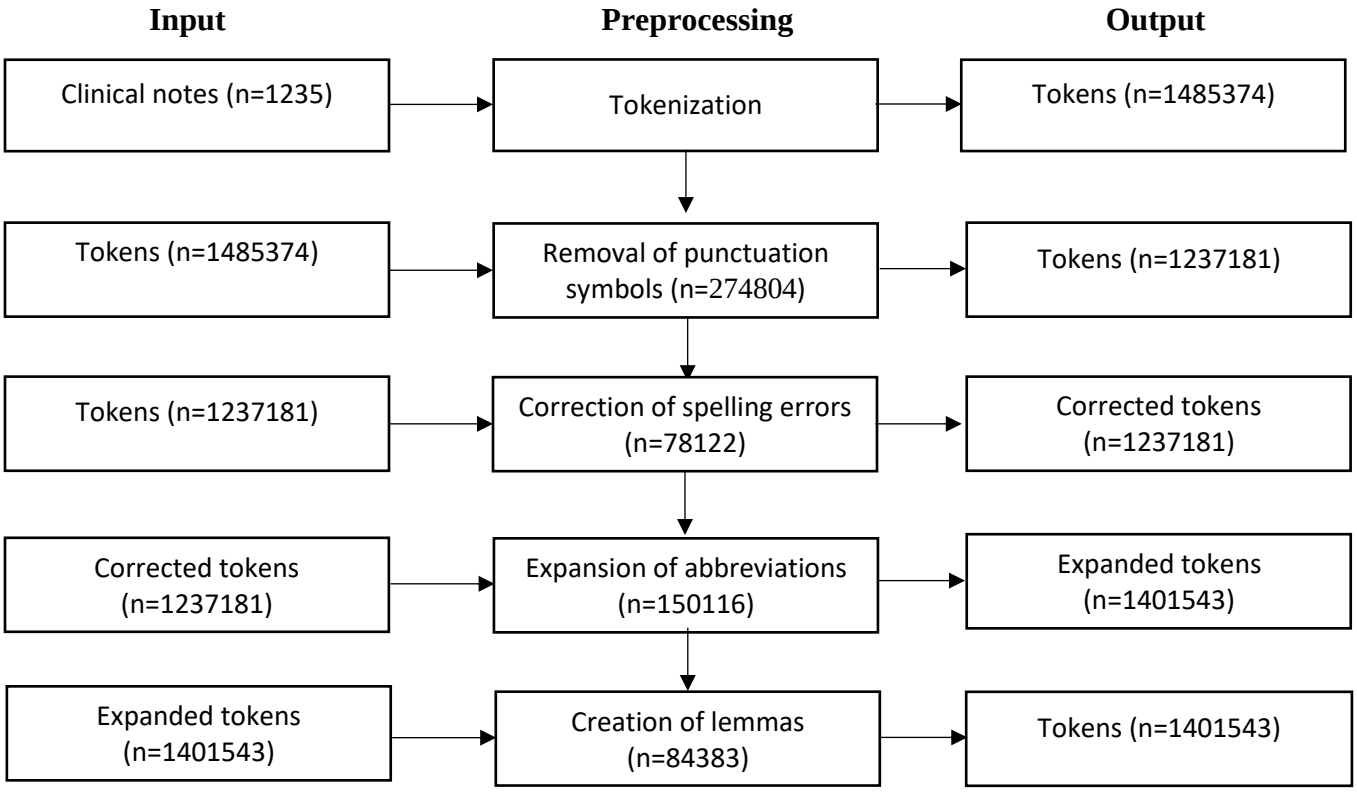


Figure 4.3: Effect of preprocessing methods in the 2008 i2b2 dataset using five preprocessing methods including lemmatization for order 1 (See Table 3.2 for the order of preprocessing methods).

Figure 4.4 shows that there were 0.90 million tokens at the beginning of preprocessing that reduced to 0.75 million tokens when four preprocessing methods were used for order 1 in the 2018 i2b2 dataset. There were 19.0% of tokens with punctuation symbols that were removed during preprocessing. After preprocessing, 6.3% of misspelled tokens were corrected.

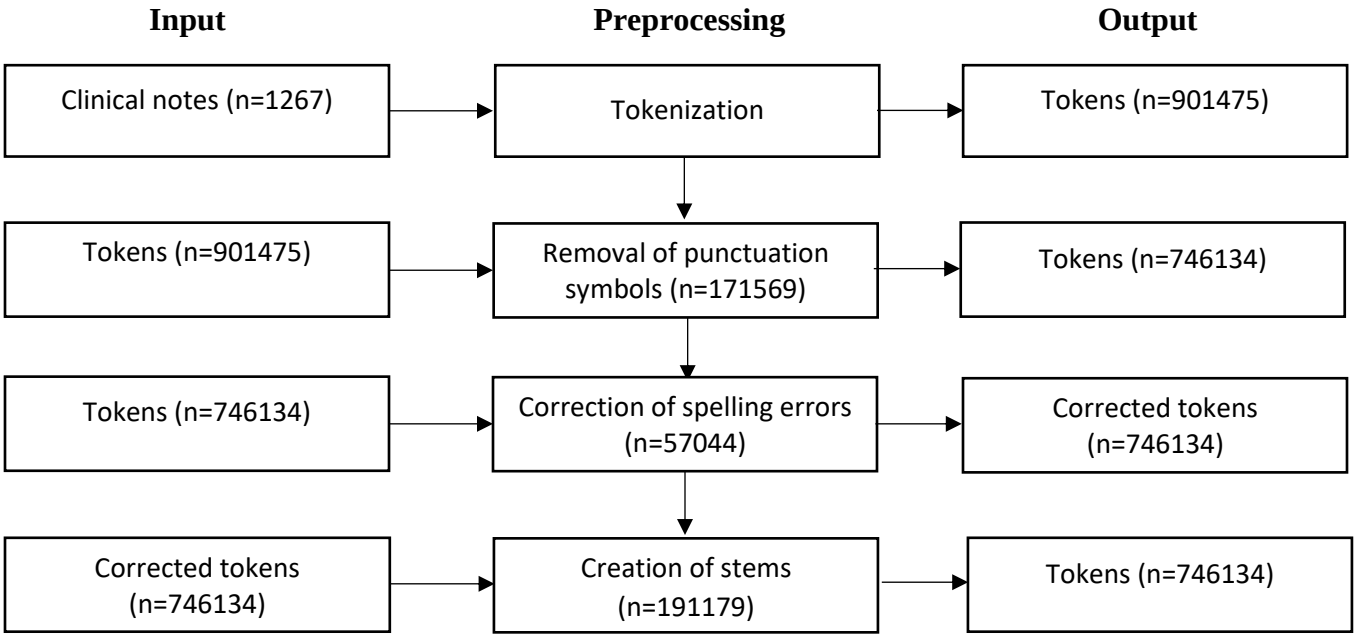


Figure 4.4: Effect of preprocessing methods in the 2018 i2b2 dataset using four preprocessing methods for order 1 (See Table 3.2 for the order of preprocessing methods).

Figures 4.5 and 4.6 show the effect of preprocessing methods when five preprocessing methods were used, including stemming and lemmatization, respectively, in the 2018 i2b2 dataset for order 1. The percentages of tokens with punctuation symbols and spelling errors were 19.0% and 6.3 %, respectively. There were 11.7% of tokens with abbreviations that were expanded during preprocessing. Though I removed punctuation symbols, the number of tokens increased because abbreviations were expanded. One-third of tokens (33.0%) were stemmed, but only 5.7% of tokens were lemmatized as shown in Figures 4.5 and 4.6, respectively.

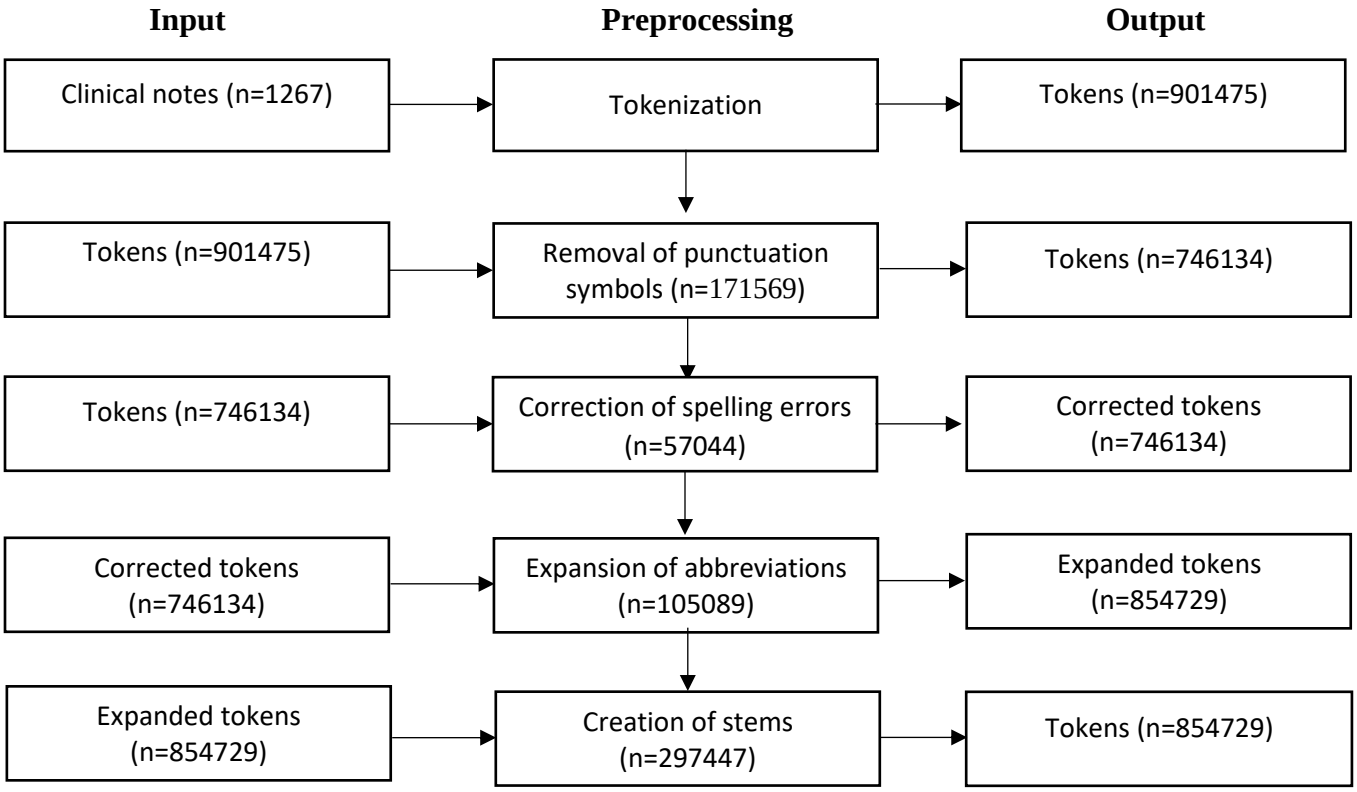


Figure 4.5: Effect of preprocessing methods in the 2018 i2b2 dataset using five preprocessing methods including stemming for order 1 (See Table 3.2 for the order of preprocessing methods).

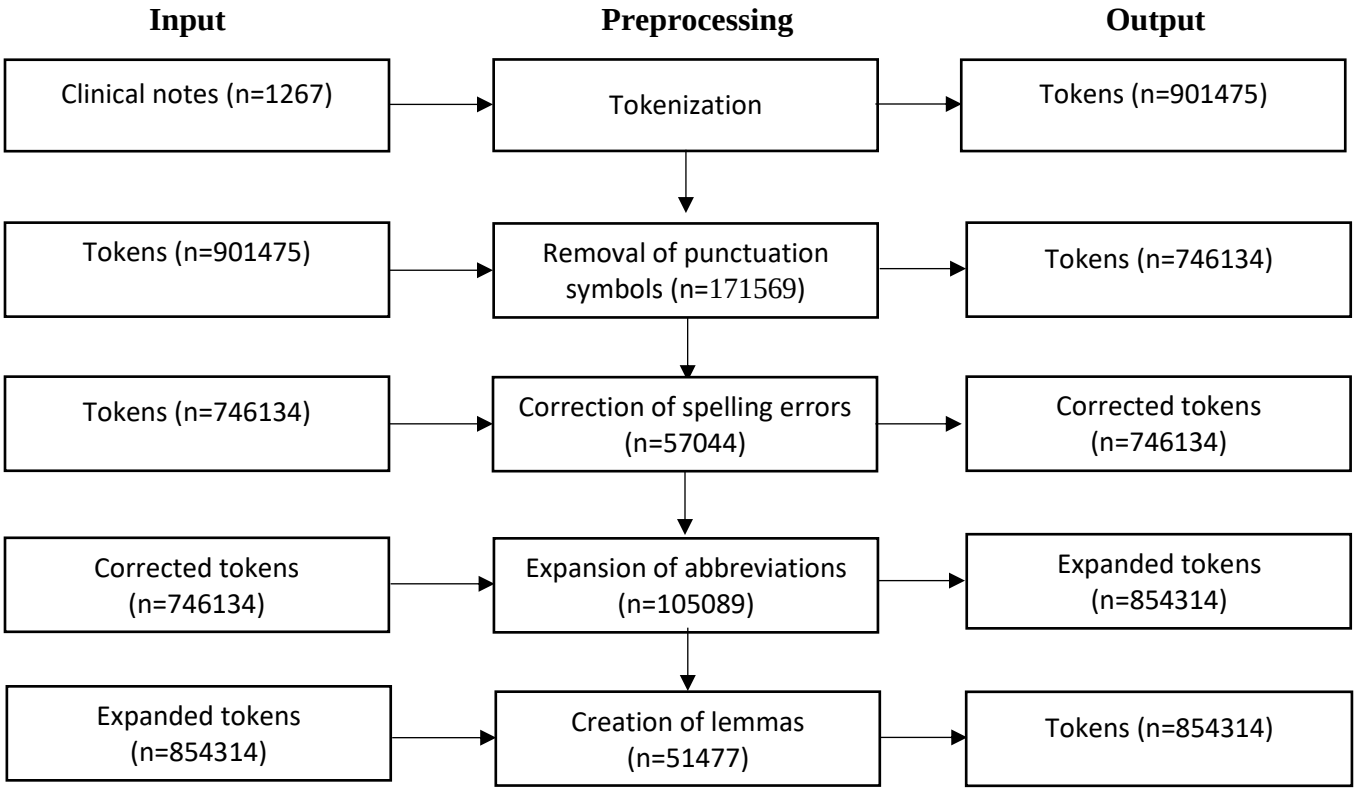


Figure 4.6: Effect of preprocessing methods in the 2018 i2b2 dataset using five preprocessing methods including lemmatization for order 1 (See Table 3.2 for the order of preprocessing methods).

4.3 Classification results

For Objective 1, I used a BRF model to detect 16 health conditions, whereas for Objective 2, I used an SVM model to identify 13 cohort selection criteria.

4.3.1 Detection of health conditions

Table 4.4 shows the estimates of the mean and SD of each of the evaluation metrics across all health conditions using the original 2008 i2b2 dataset without preprocessing. The mean estimates were 0.76 for sensitivity, 0.77 for specificity, 0.58 for F1 score, 0.76 for accuracy, and 0.53 for precision (Table 4.4). The high v SD values of 0.21 and 0.28 were for F1 score and precision, respectively. These values indicate that there were variations in the model

performance for detecting different health conditions and this might be because of class imbalance.

Table 4.4: Mean (SD) estimates of evaluation metrics for Objective 1 (detecting 16 health conditions) using the 2008 i2b2 dataset and Objective 2 (selecting cohort based on 13 criteria) using the 2018 i2b2 dataset without preprocessing.

Dataset	Sensitivity (SD)	Specificity (SD)	F1 Score (SD)	Accuracy (SD)	Precision (SD)
2008 i2b2	0.76 (0.07)	0.77 (0.08)	0.58 (0.21)	0.76 (0.08)	0.53 (0.28)
2018 i2b2	0.49 (0.40)	0.59 (0.38)	0.49 (0.37)	0.75 (0.20)	0.49 (0.35)

Note: SD = standard deviation

Table 4.5 shows the estimates of mean and SD when using four preprocessing methods with different orders. The mean values of sensitivity, specificity, F1 score, accuracy, and precision were consistent for all orders, and they were 0.76, 0.77, 0.60, 0.77, and 0.54, respectively. Table 4.5 shows that there were high values of SD for F1 score and precision.

Table 4.5: Mean (SD) estimates of evaluation metrics for Objective 1 (detecting 16 health conditions) using four preprocessing methods.

Order	Sensitivity (SD)	Specificity (SD)	F1 Score (SD)	Accuracy (SD)	Precision (SD)
1	0.76 (0.08)	0.78 (0.08)	0.60 (0.21)	0.77 (0.08)	0.54 (0.28)
2	0.76 (0.07)	0.78 (0.08)	0.60 (0.21)	0.77 (0.08)	0.54 (0.28)

Note: SD = standard deviation; See Table 3.2 for the order of preprocessing methods

The following table shows the estimates of each of the evaluation metrics for detecting 16 health conditions using the BRF model when the data were preprocessed using five preprocessing methods including word stemming (Table 4.6). The mean estimates of sensitivity were either 0.76 or 0.77 across the orders, F1 score had mean estimates of 0.60 or 0.61. The mean estimates of specificity, accuracy, and precision were identical across the orders. The estimates of evaluation metrics were slightly higher when I used five preprocessing methods compared to no use of preprocessing methods.

Table 4.6: Mean (SD) estimates of evaluation metrics for Objective 1 (detecting 16 health conditions) using five preprocessing methods including stemming.

Order	Sensitivity (SD)	Specificity (SD)	F1 Score (SD)	Accuracy (SD)	Precision (SD)
1	0.77 (0.07)	0.79 (0.09)	0.61 (0.21)	0.78 (0.08)	0.55 (0.28)
2	0.77 (0.07)	0.79 (0.08)	0.61 (0.21)	0.78 (0.08)	0.56 (0.27)
3	0.76 (0.07)	0.79 (0.08)	0.60 (0.21)	0.78 (0.08)	0.55 (0.27)
4	0.77 (0.07)	0.79 (0.08)	0.61 (0.21)	0.78 (0.08)	0.55 (0.28)
5	0.76 (0.07)	0.79 (0.08)	0.61 (0.21)	0.78 (0.08)	0.55 (0.28)
6	0.77 (0.07)	0.79 (0.08)	0.61 (0.21)	0.78 (0.08)	0.55 (0.27)

Note: SD = standard deviation; See Table 3.2 for the order of preprocessing methods

Table 4.7 shows the mean estimates of five evaluation metrics for detecting different health conditions when I used five preprocessing methods, including lemmatization. Table 4.7 shows almost similar results to that of Table 4.6.

Table 4.7: Mean (SD) estimates of evaluation metrics for Objective 1 (detecting 16 health conditions) using five preprocessing methods including lemmatization.

Order	Sensitivity (SD)	Specificity (SD)	F1 Score (SD)	Accuracy (SD)	Precision (SD)
1	0.76 (0.08)	0.79 (0.08)	0.60 (0.21)	0.78 (0.08)	0.55 (0.28)
2	0.77 (0.08)	0.79 (0.08)	0.61 (0.21)	0.78 (0.08)	0.56 (0.27)
3	0.77 (0.07)	0.79 (0.08)	0.61 (0.21)	0.78 (0.08)	0.55 (0.27)
4	0.76 (0.07)	0.79 (0.08)	0.60 (0.21)	0.78 (0.08)	0.55 (0.28)
5	0.76 (0.08)	0.79 (0.08)	0.60 (0.21)	0.78 (0.08)	0.55 (0.28)
6	0.76 (0.08)	0.79 (0.08)	0.60 (0.21)	0.78 (0.08)	0.55 (0.28)

Note: SD = standard deviation; See Table 3.2 for the order of preprocessing methods

4.3.2 Cohort selection

For Objective 2, I used an SVM model for the 13 selection criteria using the preprocessed data. Table 4.4 shows the results of evaluation metrics for selecting 13 criteria using an SVM model without preprocessing the 2018 i2b2 dataset. The mean estimates of sensitivity, specificity, F1 score, accuracy, and precision ranged from 0.49 to 0.75 with high SD values that reflect the class imbalances in the dataset.

Table 4.8 shows the results of selecting 13 cohort selection criteria using four preprocessing methods. The table shows that the mean estimates of specificity, F1 score, and accuracy were identical for different orders. The mean estimates of specificity, F1 score, and accuracy were 0.59, 0.49, and 0.75, respectively. The mean estimates of sensitivity and precision were 0.49 or 0.50 for different orders.

Table 4.8: Mean (SD) estimates of evaluation metrics for Objective 2 (selecting cohort based on 13 selection criteria) using four preprocessing methods.

Order	Sensitivity (SD)	Specificity (SD)	F1 Score (SD)	Accuracy (SD)	Precision (SD)
1	0.50 (0.38)	0.59 (0.38)	0.49 (0.36)	0.75 (0.20)	0.50 (0.35)
2	0.49 (0.39)	0.59 (0.37)	0.49 (0.37)	0.75 (0.20)	0.49 (0.35)

Note: SD = standard deviation; See Table 3.2 for the order of preprocessing methods

Table 4.9 and Table 4.10 summarize the results of evaluation metrics for selecting 13 cohort selection criteria using five preprocessing methods including stemming or lemmatization, respectively. The mean estimates of sensitivity and precision were identical (0.49) for different orders. The mean estimates of specificity ranged from 0.58 to 0.59, the F1 score ranged from 0.48 to 0.49, and accuracy ranged from 0.74 to 0.75 across orders, as shown in Table 4.9.

Table 4.9: Mean (SD) estimates of evaluation metrics for Objective 2 (selecting cohort based on 13 selection criteria) using five preprocessing methods including stemming.

Order	Sensitivity (SD)	Specificity (SD)	F1 Score (SD)	Accuracy (SD)	Precision (SD)
1	0.49 (0.39)	0.59 (0.37)	0.48 (0.37)	0.74 (0.20)	0.49 (0.35)
2	0.49 (0.39)	0.58 (0.37)	0.49 (0.37)	0.74 (0.20)	0.49 (0.35)
3	0.49 (0.39)	0.58 (0.37)	0.49 (0.37)	0.75 (0.20)	0.49 (0.35)
4	0.49 (0.39)	0.59 (0.37)	0.48 (0.37)	0.74 (0.20)	0.49 (0.35)
5	0.49 (0.39)	0.59 (0.37)	0.49 (0.37)	0.75 (0.20)	0.49 (0.35)
6	0.49 (0.39)	0.59 (0.37)	0.49 (0.37)	0.75 (0.20)	0.49 (0.35)

Note: SD = standard deviation; See Table 3.2 for the order of preprocessing methods

The mean estimates of sensitivity, specificity, F1 score, accuracy, and precision were consistent across orders shown in Table 4.10.

Table 4.10: Mean (SD) estimates of evaluation metrics for Objective 2 (selecting cohort based on 13 selection criteria) using five preprocessing methods including lemmatization.

Order	Sensitivity (SD)	Specificity (SD)	F1 Score (SD)	Accuracy (SD)	Precision (SD)
1	0.49 (0.39)	0.59 (0.37)	0.49 (0.37)	0.75 (0.20)	0.49 (0.35)
2	0.49 (0.39)	0.59 (0.37)	0.49 (0.37)	0.75 (0.20)	0.49 (0.35)
3	0.49 (0.39)	0.59 (0.37)	0.48 (0.37)	0.75 (0.20)	0.49 (0.35)
4	0.49 (0.39)	0.59 (0.37)	0.49 (0.37)	0.75 (0.19)	0.50 (0.35)
5	0.49 (0.39)	0.59 (0.37)	0.49 (0.37)	0.75 (0.20)	0.49 (0.35)
6	0.49 (0.39)	0.59 (0.37)	0.49 (0.37)	0.75 (0.20)	0.49 (0.35)

Note: SD = standard deviation; See Table 3.2 for the order of preprocessing methods

4.4 Results of statistical analyses

Table 4.11 and Table 4.12 summarize the results of the one-way ANOVA *F*- test for testing the differences among the order of methods for the 2008 and 2018 i2b2 datasets, respectively. The histograms of the evaluation metrics did not provide enough information about

the normality of the values (Figure A1-A6 in the Appendix). The difference between the mean and median was small (up to 0.14), and the estimates of skewness and kurtosis were within the cut-off values of +/- 2, and +/- 7, respectively (Tables A9 and A10 in the Appendix). It indicated that the assumption of normality was satisfied. Therefore, the assumptions of ANOVA were satisfied for testing hypotheses 1 and 2.

Tables 4.11 and 4.12 show that the F statistic values were small for each of the evaluation metrics within each number of preprocessing methods. The p-values were close to 1 or equal to 1 for all the evaluation metrics.

Table 4.11: ANOVA results for testing the differences among the order of preprocessing methods that were applied to preprocess the 2008 i2b2 dataset for Objective 1.

Number of preprocessing methods	**Sensitivity, *F* statistic (p-value)**	**Specificity, *F* statistic (p-value)**	**F1 Score, *F* statistic (p-value)**	**Accuracy, *F* statistic (p-value)**	**Precision, *F* statistic (p-value)**
4	<0.01 (0.96)	<0.01 (0.98)	<0.01 (0.98)	<0.01 (0.98)	<0.01 (1.00)
5 (with stemming)	0.01 (1.00)	<0.01 (1.00)	<0.01 (1.00)	0.02 (1.00)	<0.01 (1.00)
5 (with lemmatization)	0.13 (0.99)	0.01 (1.00)	<0.01 (1.00)	0.03 (1.00)	<0.01 (1.00)

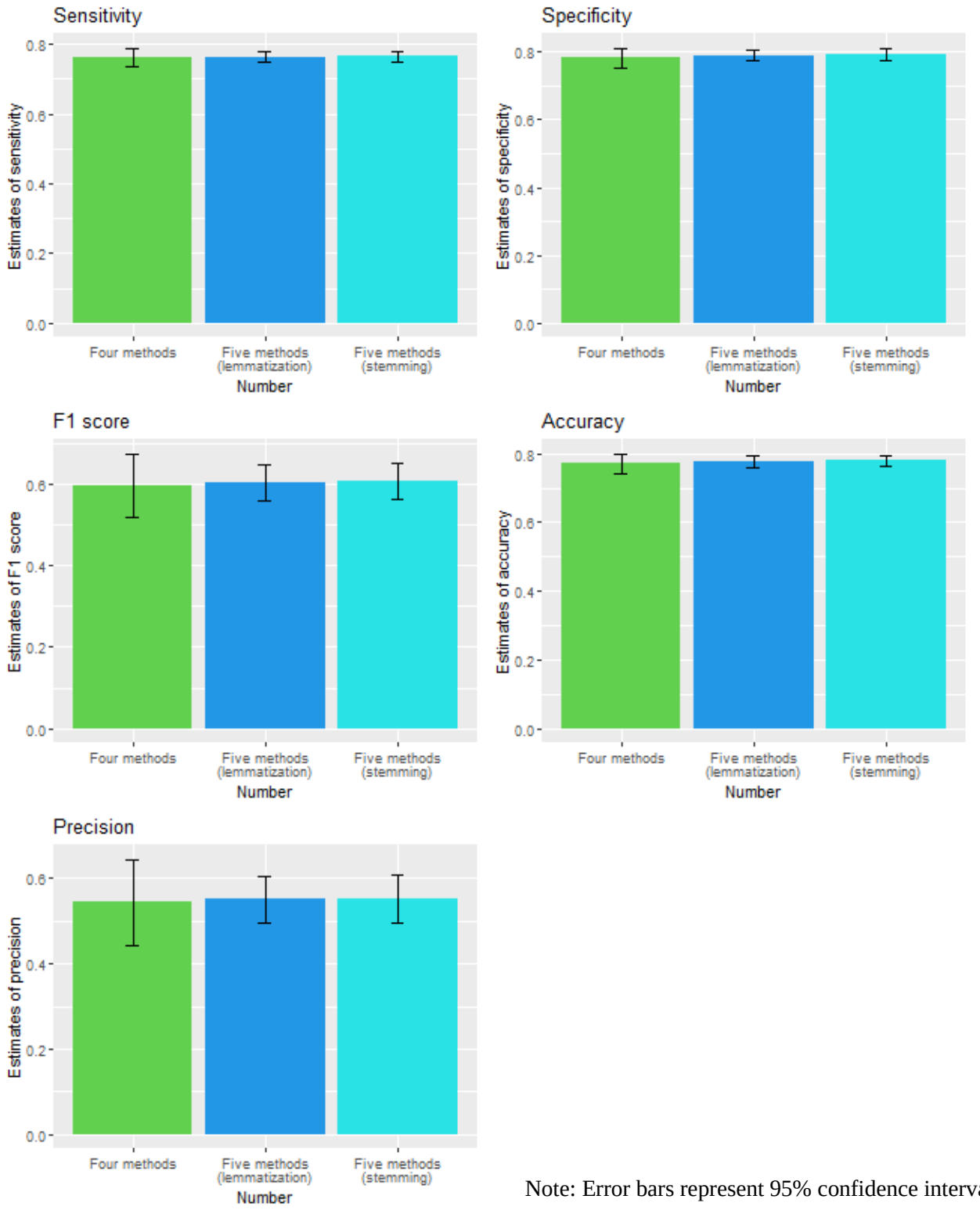
Note: degrees of freedom (df) for four preprocessing methods: numerator df = 1 and denominator df = 30; df for five preprocessing methods: numerator df = 5 and denominator df = 90

Table 4.12: ANOVA results for testing the differences among the order of preprocessing methods that were applied to preprocess the 2018 i2b2 dataset for Objective 2.

Number of preprocessing methods	**Sensitivity, *F* statistic (p-value)**	**Specificity, *F* statistic (p-value)**	**F1 Score, *F* statistic (p-value)**	**Accuracy, *F* statistic (p-value)**	**Precision, *F* statistic (p-value)**
4	<0.01 (0.95)	<0.01 (0.96)	<0.01 (0.99)	<0.01 (1.00)	<0.01 (1.00)
5 (with stemming)	0.04 (1.00)	0.01 (1.00)	<0.01 (1.00)	<0.01 (1.00)	<0.01 (1.00)
5 (with lemmatization)	<0.01 (1.00)	<0.01 (1.00)	<0.01 (1.00)	<0.01 (1.00)	<0.01 (1.00)

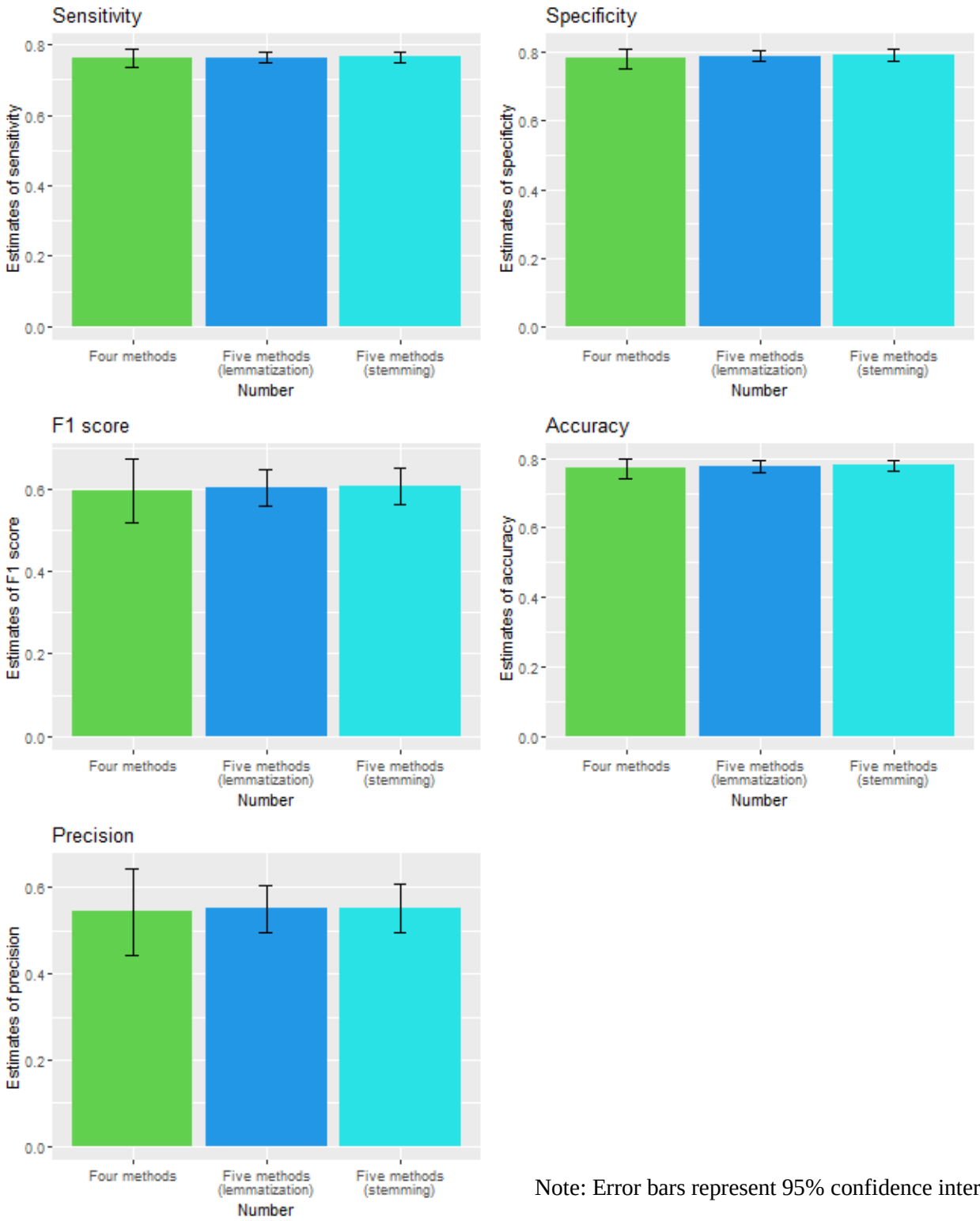
Note: degrees of freedom (df) for four preprocessing methods: numerator df = 1 and denominator df = 24; df for five preprocessing methods: numerator df = 5 and denominator df = 72.

Figure 4.7 and Figure 4.8 show the mean estimates of each of the evaluation metrics for Objective 1 (detecting 16 health conditions) and Objective 2 (identifying 13 cohort selection criteria) using different number of preprocessing methods that applied to the 2008 and 2018 i2b2 datasets, respectively. The mean values were identical for each of the evaluation metrics among the number of methods for both datasets. This indicated a low value of standard deviation among the number of methods. The standard deviations among the number of methods for sensitivity, specificity, F1 score, accuracy, and precision were 0.002, 0.005, 0.005, 0.004, and 0.005, respectively for the 2008 i2b2 dataset. The standard deviation among the number of methods for sensitivity, specificity, F1 score, and accuracy were similar (0.002) for the 2018 i2b2 dataset. The standard deviation among the number of methods for precision was 0.004.



Note: Error bars represent 95% confidence intervals

Figure 4.7: Mean estimates of evaluation metrics for Objective 1 (detecting 16 health conditions) using different numbers of preprocessing methods.



Note: Error bars represent 95% confidence intervals

Figure 4.8: Mean estimates of evaluation metrics for Objective 2 (identifying cohort selection criteria) using different numbers of preprocessing methods.

4.5 Summary

The preprocessing of the UTD from EMR reduced the size of the datasets by removing noises from the notes, especially when I used word stemming instead of lemmatization. There were a lot of punctuation symbols that were redundant for analysis were removed from both datasets as well as there were many spelling errors and abbreviations in both datasets that were corrected/expanded during preprocessing. The performances of the models were very similar for different order and number of preprocessing methods for both datasets, respectively. The values of evaluation metrics were not significant at 5% level of significance among the orders. The identical mean estimates and the small standard deviation implied that there was no difference among the number of methods for each of the evaluation metrics for both datasets. There were high values of SD among F1 score and precision due to the class imbalances in the 2008 i2b2 dataset. Since there was class imbalance in the 2018 i2b2 dataset for most of the criteria, we had high SD values for each of the evaluation metrics.

Chapter 5: Results for similarity of text-based measures

5.1 Data characteristics

I selected the first two notes from each of the patients (each patient had 2 to 5 notes) to create a pair of notes for measuring the similarity between pairs of EMR notes from the 2018 i2b2 dataset. The notes of a patient might have come from different physicians; this was not assessed. This dataset had 576 notes from 288 patients.

Table 5.1: Characteristics of the 2018 i2b2 dataset that was used for measuring similarity.

Attributes	Frequency	Mean (SD) per note	Median (Q1, Q3) per note
Clinical notes	576	-	-
Tokens	385618	669.5 (405.4)	553.0 (397.8, 837.3)
Punctuation symbols	73759	128.1 (85.4)	105.0 (75.0, 153.0)
Spelling errors	25917	45.0 (42.7)	30.0 (16.0, 56.3)
Abbreviations	44623	77.5 (58.3)	61.0 (38.0, 98.3)

Note: SD = Standard deviation; Q1 = 1st quartile, Q3 = 3rd quartile

Table 5.1 shows that the dataset had a mean (SD) of 669.5 (405.4) tokens per note. The dataset had a mean (SD) of 128.1 (85.4) punctuation symbols, 45.0 (42.7) spelling errors, and 77.5 (58.3) abbreviations per note.

5.2 Effect of preprocessing methods

Figure 5.1 shows the effect of preprocessing methods when four preprocessing methods were applied to preprocess the dataset of pairs of notes for order 1. The total number of tokens was higher at the beginning of preprocessing compared to the total number of tokens at the end of preprocessing. There were 18.6% of tokens with punctuation symbols from which punctuation symbols were removed. Misspelled tokens were 5.7% that were corrected during preprocessing. Word stemming stemmed 21.9% tokens.

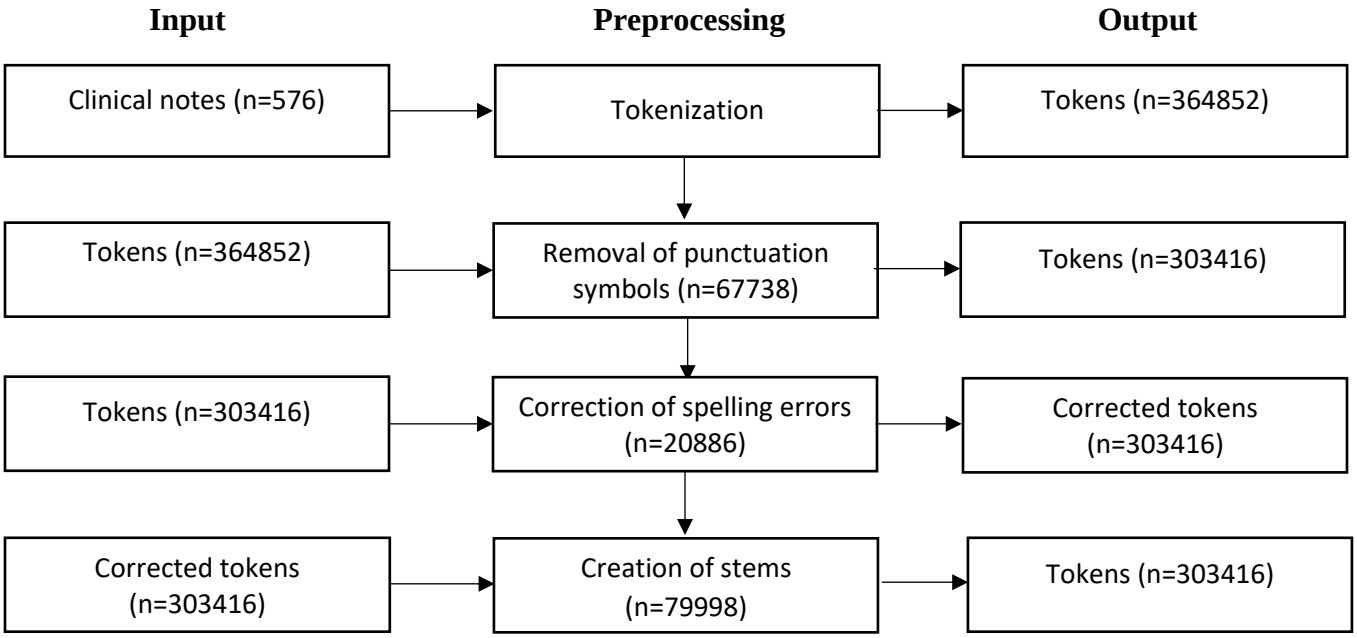


Figure 5.1: Effect of preprocessing methods in the subset of the 2018 i2b2 dataset using four preprocessing methods for order 1 (See Table 3.2 for the order of preprocessing methods).

Figure 5.2 and Figure 5.3 demonstrate the effect of preprocessing methods in the dataset using five preprocessing methods including stemming and lemmatization, respectively. Since I tokenized the data first during preprocessing, the number of tokens at the beginning was same all the time. The punctuation symbols and spelling errors were 18.6% and 5.7%, respectively. There were 11.1% of abbreviated tokens in both figures that were expanded during preprocessing. A large number of tokens (33.1%) were stemmed, whereas only 6.0% of tokens were lemmatized.

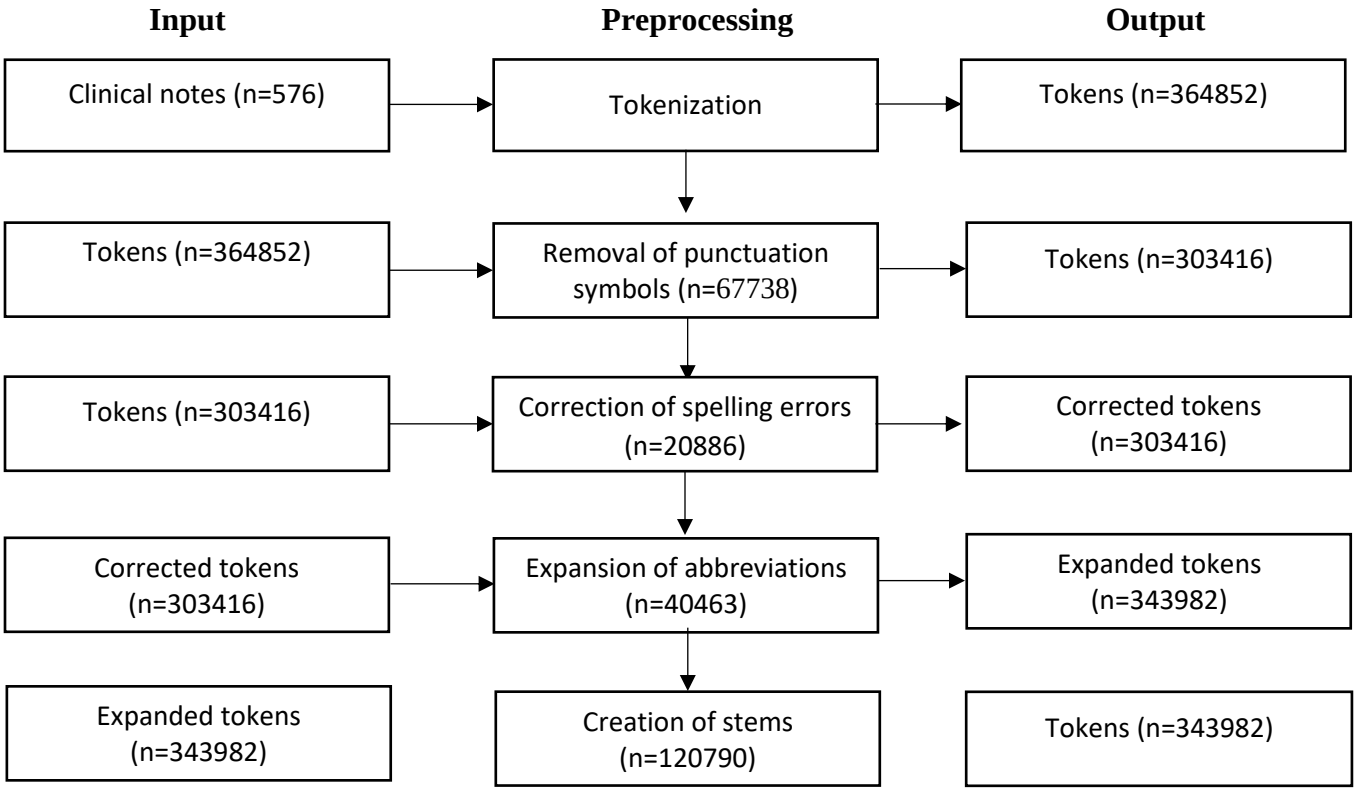


Figure 5.2: Effect of preprocessing methods in the subset of the 2018 i2b2 dataset using five preprocessing methods including stemming for order 1 (See Table 3.2 for the order of preprocessing methods).

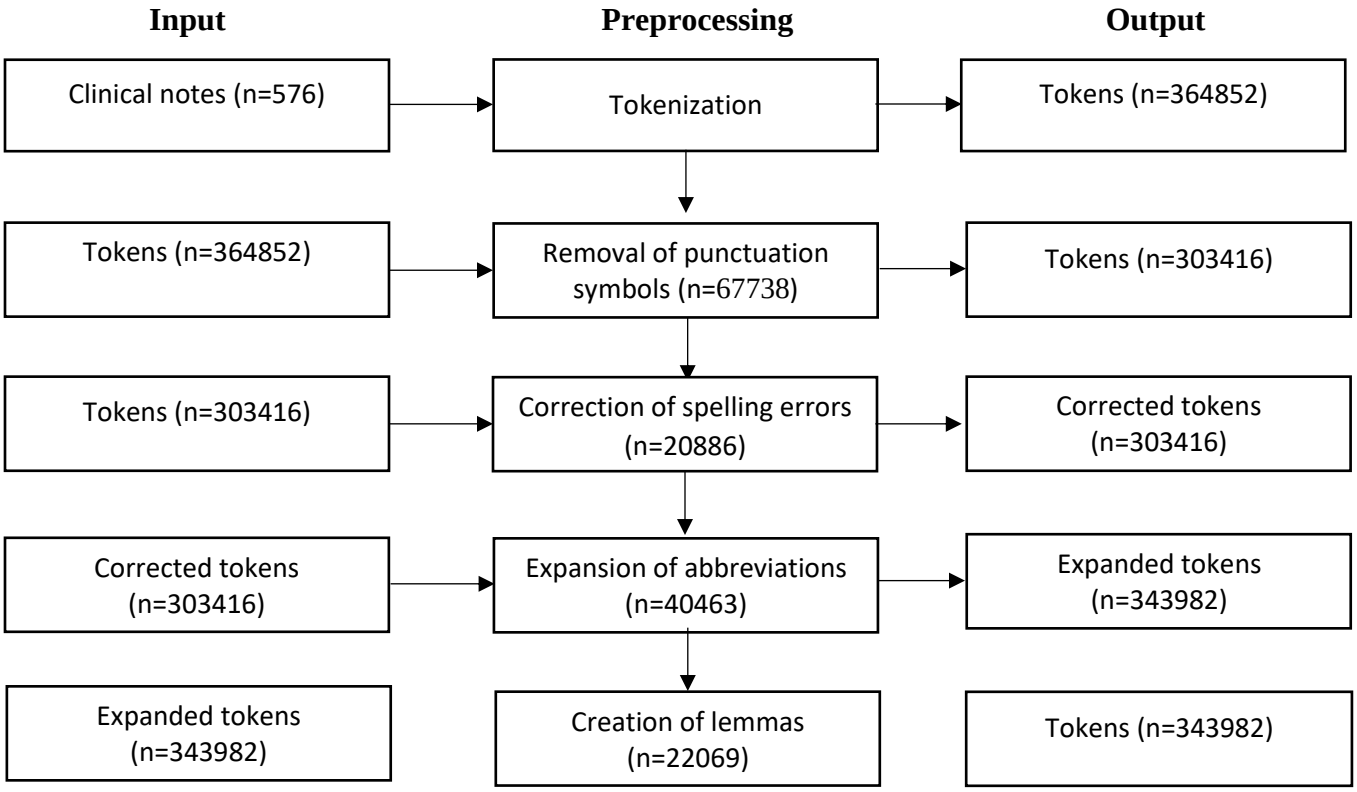


Figure 5.3: Effect of preprocessing methods in the subset of the 2018 i2b2 dataset using five preprocessing methods including lemmatization for order 1 (See Table 3.2 for the order of preprocessing methods).

5.3 Results for similarity measures and statistical analyses

For Objective 3 to assess the effect of order and number of preprocessing methods on the similarity of information, I estimated the similarity of information between pairs of EMR notes using cosine similarity and reported the mean and SD of cosine similarities across all the pairs of notes. The mean (SD) similarity estimate was 0.47 (0.17) when the dataset was not preprocessed. Table 5.2 shows the results of the mean of cosine similarities among the pairs of notes for different orders and the number of preprocessing methods that were applied to the dataset. The mean estimate of cosine similarities was identical (0.49) for all orders when four preprocessing methods were applied. The mean estimate of cosine similarities was either 0.49 or 0.50 when five preprocessing methods with stemming were applied to the dataset. The mean value of cosine

similarities was identical (0.49) for all orders when I used five preprocessing methods with lemmatization.

The histograms did not provide enough information about the normality of cosine similarities (Figure A7 in the Appendix). There was no difference between the mean and median when using four and five preprocessing methods with stemming, and the difference was small 0.01 for five preprocessing methods with lemmatization (Table A11 in the Appendix). The values of skewness were very small, and the values of kurtosis were within the cut-offs of +/- 7. Therefore, the assumption of normality was satisfied. Since the variance was constant and the normality assumption was satisfied, an ANOVA *F* test was carried out to test the effect of the order of preprocessing methods on the similarity of EMR notes for the same individual.

The results show that the *F* statistic values were very small for each number of preprocessing methods (Table 5.2). The p-values were close to 1 or equivalent to 1 (Table 5.2).

Table 5.2: Mean (SD) estimates of cosine similarities and results of ANOVA for testing differences among the order of preprocessing methods that were applied to preprocess the subset of the 2018 i2b2 dataset for Objective 3.

Order	**Number of preprocessing methods**					
	4		**5 (with stemming)**		**5 (with lemmatization)**	
	Mean (SD)	*F* statistic (p-value)	Mean (SD)	*F* statistic (p-value)	Mean (SD)	*F* statistic (p-value)
1	0.49 (0.17)		0.49 (0.16)		0.49 (0.17)	
2	0.49 (0.17)		0.50 (0.17)		0.49 (0.17)	
3	-	<0.01	0.49 (0.16)	0.04	0.49 (0.17)	0.02
4	-	(0.95)	0.50 (0.16)	(1.00)	0.49 (0.16)	(1.00)
5	-		0.50 (0.17)		0.49 (0.17)	
6	-		0.50 (0.17)		0.49 (0.17)	

Note: SD = standard deviation; See Table 3.2 for the order of preprocessing methods; degrees of freedom (df): numerator df = 1 and denominator = 574 for four preprocessing methods and numerator df =1, and denominator df = 1722 for five preprocessing methods

The mean estimates of cosine similarity scores were identical, and the standard deviation was 0.005. The identical mean estimates and the small standard deviation implied that there was no difference among the number of methods in terms of cosine similarity.

5.4 Summary

The size of the dataset was reduced due to the preprocessing of the UTD. The mean of cosine similarities increased because of the preprocessing of the UTD compared to no preprocessing. When we preprocessed the data with five preprocessing methods including stemming, the mean cosine similarity values increased. The mean cosine similarity scores were identical when using four and five preprocessing methods including lemmatization. However, the effect of the preprocessing methods among the orders was not statistically significant. There was no difference among the number of methods found by analyzing descriptive statistics.

Chapter 6: Results for de-identification of UTD

6.1 Data characteristics

For Objective 4, to de-identify the UTD from EMR data, we used the 2014 i2b2 dataset. The dataset had 1304 clinical notes for 296 patients. Table 6.1 shows that the dataset had a mean (SD) of 749.9 (435.8) tokens per note. The dataset had a mean (SD) of 147.4 (101.2) punctuation symbols, 54.8 (46.4) spelling errors, and 92.6 (64.7) abbreviations per note.

Table 6.1: Characteristics of the 2014 i2b2 dataset.

Attributes	Frequency	Mean (SD) per note	Median (Q1, Q3) per note
Clinical notes	1304	-	-
Tokens	977908	749.9 (435.8)	638.0 (435.0, 988.0)
Punctuation symbols	192234	147.4 (101.2)	119.0 (84.0, 182.3)
Spelling errors	71521	54.8 (46.4)	40.0 (20.0, 77.0)
Abbreviations	120809	92.6 (64.7)	74.0 (44.0, 126.0)

Note: SD = Standard deviation; Q1 = 1st quartile, Q3 = 3rd quartile

6.2 Effect of preprocessing methods

Figure 6.1 shows the effect of using four preprocessing methods for order 1. Since tokenization was performed at the beginning during preprocessing, the number of tokens (0.98 million) was the same for all combinations of preprocessing methods. The number was reduced to 802 thousand at the end of preprocessing using four preprocessing methods. There were 19.6% of tokens with punctuation symbols that were removed and 6.4% of tokens were misspelled that were corrected after preprocessing. One-fifth of the total tokens were stemmed.

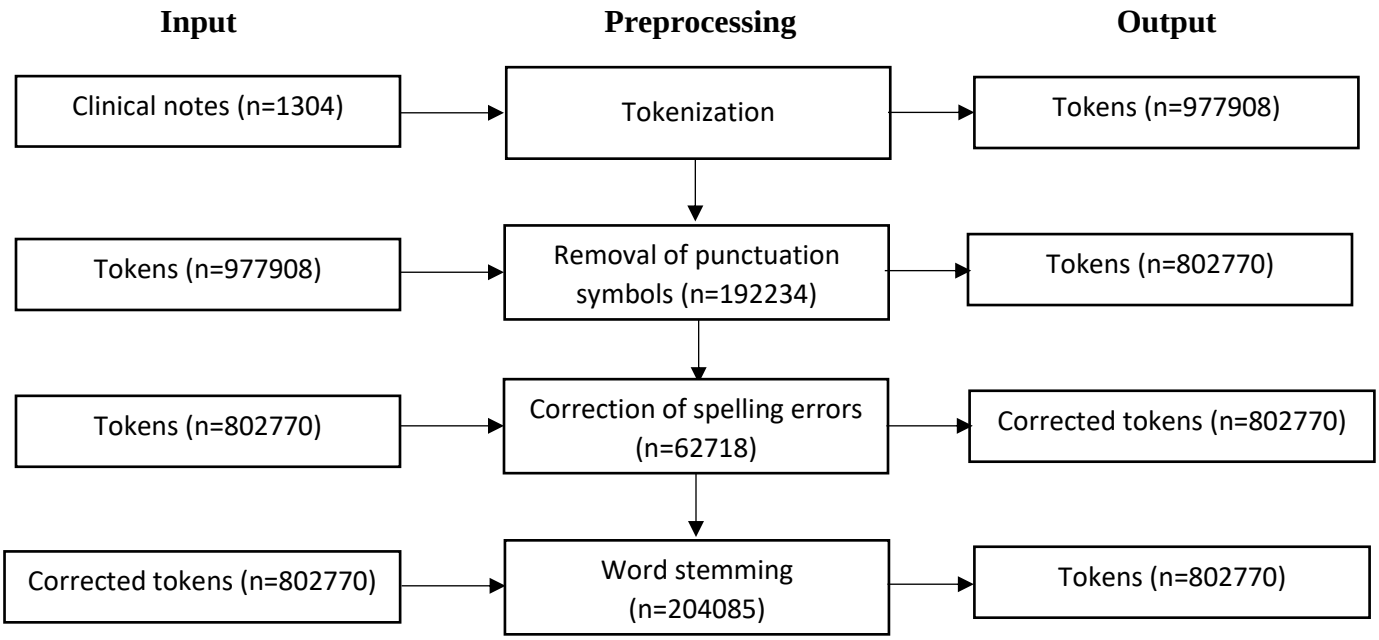


Figure 6.1: Effect of preprocessing methods in the 2014 dataset using four preprocessing methods for order 1 (See Table 3.2 for the order of preprocessing methods).

Figure 6.2 and Figure 6.3 show the effect of different preprocessing methods using five preprocessing methods including word stemming and lemmatization, respectively. The number of tokens was similar to the number of tokens when using four preprocessing methods. The percentages of tokens with punctuation symbols and spelling errors were 19.6% and 6.4%, respectively. There were 11.5% of tokens with abbreviations that were expanded after preprocessing, as shown in Figure 6.2 and Figure 6.3. One-third of tokens were stemmed during preprocessing, whereas only 5.5% of tokens were lemmatized during preprocessing.

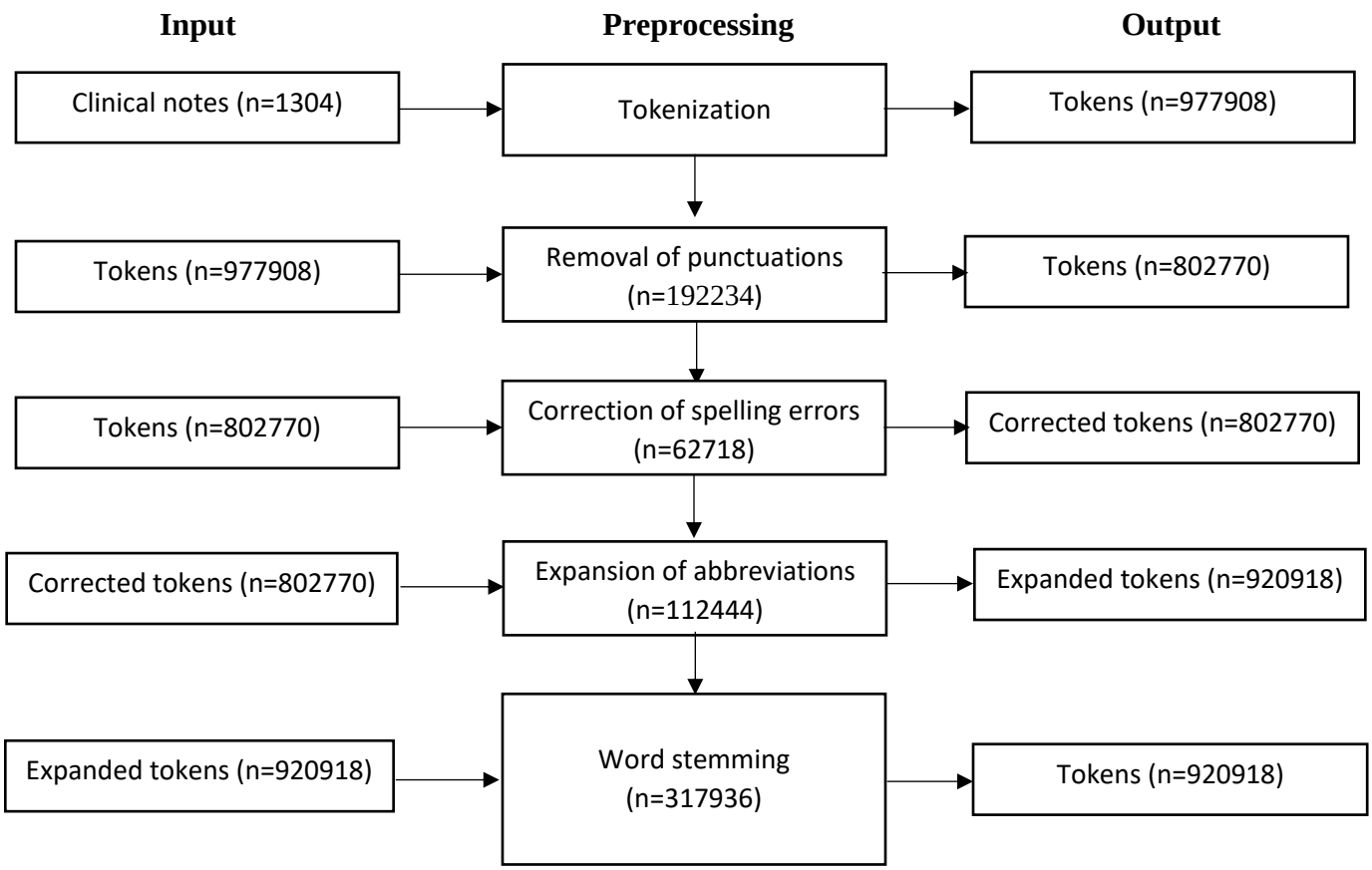


Figure 6.2: Effect of preprocessing methods in the 2014 dataset using five preprocessing methods including stemming for order 1 (See Table 3.2 for the order of preprocessing methods).

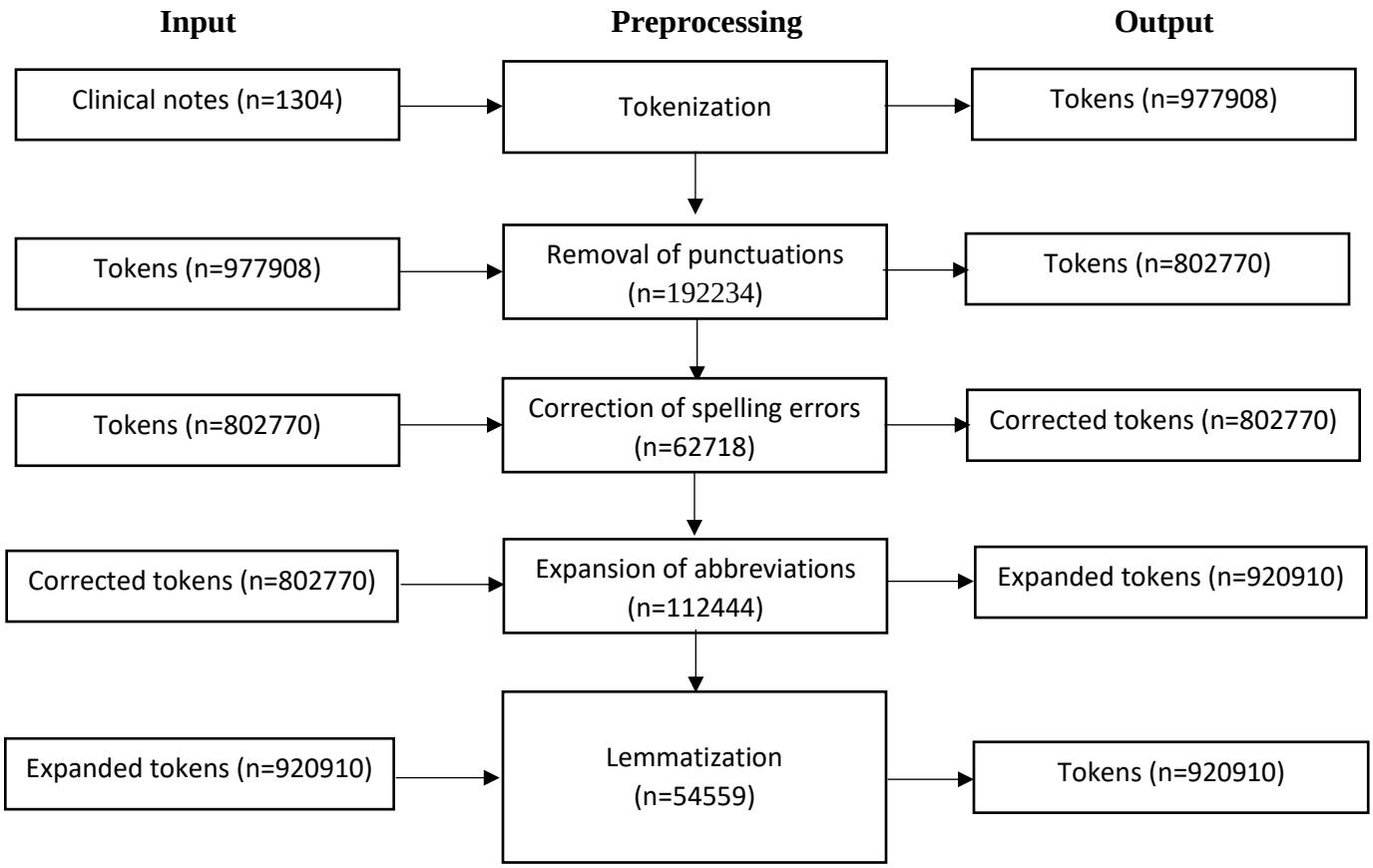


Figure 6.3: Effect of preprocessing methods in the 2014 dataset using five preprocessing methods including lemmatization for order 1 (See Table 3.2 for the order of preprocessing methods).

6.3 Challenges with model implementation

There were challenges associated with the implementation of the proposed de-identification model using the preprocessed data. The Bi-LSTM and CRF model worked well when the original 2014 i2b2 dataset without any preprocessing was used. However, when the preprocessed data were used, the model did not work. I found that the model required text data with sentence boundaries as input and it performed sentence segmentation by identifying each sentence and tokenization itself. I tried other preprocessing methods [e.g., removing punctuation symbols (except the period so that the model can identify the sentence), and spelling correction] to prepare the data and provided these data as input in the model. The model still did not work. I

reached out to the developer of the model, and he mentioned that we might need to adjust the annotation offsets of the PHI. During preprocessing, the start and end positions of each instance of PHI changed relative to their positions in the annotated files. To use the preprocessed data in the model, preprocessing of the annotated files was required and accordingly adjusting the offsets; this was a complex task. In addition, the author also mentioned that it is unusual to remove punctuation symbols in NER because these preprocessing steps would likely lead to incomplete or ungrammatical sentences.

Similar problems as mentioned above occurred with other deep learning models. All the deep learning based de-identification models require text data with the same start and end positions according to the annotated file because these positions are used in training the model to de-identify the UTD.

6.4 Summary

Preprocessing of the 2014 i2b2 dataset was performed using the proposed preprocessing methods and it reduced the size of the data as well as corrected misspelled words and expanded abbreviations. However, I was not able to use these preprocessed data in the deep learning model since deep learning models did not require much preprocessing to perform well. Most of the models only require sentence identification and tokenization as preprocessing steps; these are often built-in functions in the model that are not in the user's control.

Chapter 7: Discussion and Conclusions

7.1 Summary

This research was conducted to investigate the effect of preprocessing methods on the analysis of UTD in EMRs. The objectives were to assess the effect of the number and order of preprocessing methods for UTD on the detection of health conditions and cohort selection criteria, on the similarity of information contained in pairs of EMR notes from the same individual, and on accurate de-identification of UTD. Three publicly available i2b2 clinical challenge datasets were used in the study. The study adopted a nested experimental design to test the research hypotheses, in which the independent variables were the number and order of preprocessing methods; the orders were nested within the number of preprocessing methods. The outcome variables included several measures of predictive performance, as well as cosine similarity.

Different supervised machine learning models and cosine similarity were used to achieve the objectives. The BRF model was used to detect different health conditions from preprocessed UTD using the 2008 i2b2 dataset. An SVM model was used to identify 13 cohort selection criteria from the 2018 i2b2 dataset. The study was unable to use a deep learning model to de-identify the PHI after preprocessing the UTD due to the limitation of using preprocessed data in the model. To measure the similarity of information between pairs of EMR notes from the same patient, I measured cosine similarity. The ANOVA *F* test was used to test the differences among the order of preprocessing methods at each level of number of methods. Descriptive statistics were used to describe differences for each number of methods.

The overall results show that the values of the evaluation metrics including sensitivity, specificity, accuracy, F1 score, and precision were not statistically significant across different

orders of preprocessing methods. Though there were no statistically significant differences in the values of cosine similarities using different orders and numbers of preprocessing methods, the values of cosine similarities increased when I increased the number of preprocessing methods. There were limitations to preprocess the UTD for de-identification before implementing a deep learning model. There was no significant effect of the order of preprocessing methods in detecting health conditions, selecting cohort members, or measuring the similarity of health records.

7.2 Discussion of key findings

The 2008 i2b2 dataset was used to detect 16 health conditions and the following methods were applied to preprocess the dataset: tokenization, removal of punctuation symbols, correction of spelling errors, expansion of abbreviations, word stemming, and lemmatization. The dataset had 18.5% of tokens associated with punctuation symbols, 5.3% of tokens were misspelled, and 10.1% of tokens were abbreviations. For Objective 1, to assess the effect of preprocessing methods on the detection of obesity and its comorbidities, a BRF model was used since the class sizes were imbalanced for most of the health conditions. When the raw data were used to detect the health conditions, the mean sensitivity estimate for the BRF model was 0.76. When the preprocessed data were used to detect the health conditions, the mean sensitivity was almost identical and ranged from 0.76 to 0.77 for all orders with different numbers of preprocessing methods. The mean specificity, F1 score, accuracy, and precision increased by 1% when five preprocessing methods were used instead of four preprocessing methods. The model performed slightly better when using preprocessed data compared to raw data in terms of specificity, F1 score, accuracy, and precision. The ANOVA results for testing differences of mean estimates of sensitivity, specificity, F1 score, accuracy, and precision among the orders of preprocessing

methods showed that there was no statistically significant effect of the order of methods irrespective of the number of methods. Descriptive statistics revealed no difference among the number of preprocessing methods for each of the evaluation metrics.

For Objective 2 for selecting the cohort selection criteria, I preprocessed the 2018 i2b2 data using the following preprocessing methods: tokenization, removal of punctuation symbols, correction of spelling errors, expansion of abbreviations, word stemming, and lemmatization. A total of 19.0% of tokens of punctuation symbols were removed, 6.3% of tokens were corrected, and 11.7% of tokens were expanded during the preprocessing of the data for order 1 for each level of the number of methods. The size of the data was reduced more during stemming than lemmatization. Thirteen cohort selection criteria were identified using the SVM model (95). The mean estimate of sensitivity for the raw data was 0.49, whereas the mean estimate ranged from 0.49 to 0.50 when the preprocessed data were used for classification. The mean specificity estimates were 0.58 to 0.59, F1 score values were 0.48 to 0.49, accuracy estimates were 0.74 to 0.75, and precision estimates were 0.49 to 0.50; the corresponding estimates were 0.59, 0.49, 0.75, and 0.49 when the SVM model was applied to the raw data. The results show that there was no significant improvement in the model performance when the UTD was preprocessed with different orders and numbers of preprocessing methods. When I used lemmatization, the results were more constant compared to using stemming for five preprocessing methods. The ANOVA results showed that there were no statistically significant differences in the performance of the SVM model for different orders of methods for each level of number of methods. Regardless of whether four or five preprocessing methods were used, model performance was similar.

For Objective 3, which focused on measuring the similarity of information in EMR notes, the dataset was a subset of the 2018 i2b2 longitudinal data containing a pair of clinical notes for

each patient. There were almost 700 tokens per note, including 19.1% punctuation symbols, 6.7% spelling errors, and 11.6% abbreviations among all the tokens. The size of the data was reduced more during stemming compared to lemmatization.

When cosine similarity was used to measure the similarity of information without preprocessing, the overall mean estimate was 0.47. But when I preprocessed the UTD using the specified orders and numbers of preprocessing methods, the overall mean estimates of similarity were slightly higher. The cosine similarity scores were identical for each order of four and five (with lemmatization) preprocessing methods. Cosine similarity scores ranged from 0.49 to 0.50 when using 5 preprocessing methods with word stemming. This suggests that there were no differences in model performances among the order of methods for each level of number of preprocessing methods. The ANOVA results showed that the effect of the order of preprocessing methods was not statistically significant. The mean estimates of cosine similarity scores for each level of the number of methods were identical and the standard deviations were small.

For Objective 4, which focused on the effect of preprocessing methods on de-identification of UTD, the 2014 i2b2 data were preprocessed. The data had approximately 1 million tokens; 19.7% were punctuation symbols, 7.3% were spelling errors, and 12.4% were abbreviations. The data had more corrected and expanded tokens with smaller sizes at the end of preprocessing compared to no preprocessing. However, the study found that the deep learning models were not trainable with the preprocessed data. Some of the proposed preprocessing methods are not usually used for NER tasks. For example, punctuation symbols may contain information regarding the end of a sentence. Moreover, deep learning models may include self-developed preprocessing steps that require the raw text data with annotations.

Preprocessing reduced the size of each of the datasets by removing noise. Model performance slightly increased using preprocessed data compared to using raw data for Objectives 1 (detection of health conditions), 2 (identification of cohort selection criteria), and 3 (similarity measures); this finding is similar to the findings of prior research (3,12,14,16,21,44). A few of the studies found that the number and order of preprocessing methods could improve the classification model performance or the similarity measure (16,21). These studies used short text data from Twitter to examine the effect of preprocessing methods. However, the results of this study show that the order and number of preprocessing methods did not have any effect on model performance.

7.3 Strengths and limitations

There are some research limitations. One limitation is that the study used only a few preprocessing methods among many potential methods to process medical text data. Moreover, the number of methods had limited variation (i.e., four versus five). Though the removal of stop words and numbers are commonly used preprocessing methods for text classification, there are contradictions among researchers about the removal of stop words and numbers from the text. Some researchers have suggested removing stop words and numbers (3,16,18), whereas other researchers have suggested the converse (18,21,107). Initially, I removed stop words for Objectives 1, 2, and 3. Although the mean estimates of evaluation metrics were similar for Objectives 1 and 2, the mean estimates of cosine similarity scores decreased to 49% due to the removal of stop words compared to the mean estimates with raw data for Objective 3 on similarity measure. Removal of numbers from text has also been proposed as a preprocessing method. However, I did not consider the removal of numbers since numbers in medical records may contain information about health conditions. For example, obesity could be deduced from the numeric value of BMI, from the numeric value of weight and height, or the numeric value of

weight alone (76). Moreover, there were two criteria (e.g., creatinine and HBA1C) that were value-dependent in the 2018 i2b2 dataset. For example, do the patient records include laboratory test results with any HbA1c value between 6.5% and 9.5%?

The study used the string library in python with a customized function (not removing any punctuation symbols within a word) to remove the punctuation symbols. There are many spell-checking libraries in python and each of them might use different algorithms for spelling correction. However, I used the pyspellchecker library which uses the Levenshtein Distance (difference between two strings) algorithm to correct the misspelling. The study used a dictionary with medical terminology from GitHub because the pyspellchecker library did not have medical terminology in its built-in dictionary (49). Though there are a few libraries in python for the expansion of abbreviations, I wrote a function for the expansion of abbreviations that used a list of 3025 medical abbreviations which is a part of a python library (MEDIALpy) (47). Whenever there were two or more expansions of an abbreviation, the study randomly chose one of the expansions. There are many rule-based, machine learning, and deep learning methods that are used to detect health conditions, identify cohort selection criteria, and de-identify the UTD. Since the research goal was to assess the effect of the order and number of preprocessing methods using automated models, the research was not focused on developing new models for the study. Though deep learning models are used to de-identify the UTD and provide better results compared to rule-based and machine learning-based models, there is very limited scope to preprocess the data to use in these models. Most of the deep learning models require text data as input and preprocess the data itself using built-in functions (99–102). This limited the use of preprocessed data for de-identification in the study.

Despite these limitations, the research has many strengths. The first strength of the research is that it assessed the effect of the order and number of preprocessing methods to prepare the UTD from EMRs for health research. The research examined the effect of preprocessing methods in different tasks such as detecting different health conditions (sixteen comorbidities) and selecting cohort for clinical trial based on 13 selection criteria from the UTD. It used two different supervised machine learning models for two different classification tasks. In addition, the study estimated the similarities between pairs of EMR notes for the same patient. Furthermore, the research explored whether there was any effect of preprocessing of UTD for de-identification of PHI. Lastly, this study is an addition to the limited research on the effect of preprocessing of UTD from EMRs, which are potentially a rich source of data for research about population health and health service use.

7.4 Significance

Though UTD in EMR data provides useful information for population health and health services research, one of the challenges is variation in UTD quality. The quality of UTD from EMRs is crucial for conducting chronic disease and health services research. This study provides information about the systematic effect of the order and number of preprocessing methods on common analyses applied to UTD. Preprocessing of UTD is required to improve the data quality that may affect model performance. The study has found that the order and number of preprocessing methods did not affect model performance.

7.5 Recommendations for future research

There are several recommendations for future research about the quality of UTD from EMRs. First, the choice of spelling correction libraries and medical dictionaries might affect model performance. There are many spelling correction libraries available in python that use

different algorithms to correct misspelled words, and many medical dictionaries are available to correct misspelled words. Model performance could be compared across different spelling correction libraries and medical dictionaries. Second, many medical abbreviation lists with expansions are also available. When expanding medical abbreviations, the study could use the context to replace the abbreviation with an appropriate expansion, instead of a random choice of one expansion. Model performance for classification tasks could be compared using different abbreviation lists.

Lastly, different machine learning models could be used for classification tasks such as detecting different health conditions using UTD from EMR data, but they did not assess the effect of the order and number of preprocessing methods (3,95,108). A study could be carried out to explore the effect of the order and number of preprocessing methods in different supervised machine learning models (e.g., RF, SVM, decision tree, naïve Bayes, lasso regression model, logistic regression).

7.6 Conclusions

The research preprocessed publicly available datasets containing UTD and measured model performance or similarity of pairs of health records. The results show that preprocessing reduced the size of the data. There was no difference among the orders and numbers of preprocessing methods for classification or similarity measure. Sometimes the performance of the models or similarity measures improved slightly if expansion of abbreviations was used in addition to other preprocessing methods to prepare the UTD. The choice of python libraries to preprocess the UTD might have an effect on model performance. There was limited scope to preprocess the UTD while using an automated deep learning model to de-identify PHI from the

UTD in EMRs. Future research could investigate the effect of the source of spelling correction libraries, medical dictionaries, and abbreviation lists.

References

1. Birtwhistle R, Williamson T. Primary Care Electronic Medical Records: A New Data Source for Research in Canada. CMAJ. 2015 Mar 3;187(4):239–40.

2. Sun W, Cai Z, Liu F, Fang S, Wang G. A Survey of Data Mining Technology on Electronic Medical Records. In: 2017 IEEE 19th International Conference on e-Health Networking, Applications and Services (Healthcom). 2017. p. 1–6.

3. Mahendra M, Luo Y, Mills H, Schenk G, Butte AJ, Dudley RA. Impact of Different Approaches to Preparing Notes for Analysis with Natural Language Processing on the Performance of Prediction Models in Intensive Care. Crit Care Explor. 2021 Jun 11;3(6):e0450.

4. Marafino BJ, Park M, Davies JM, Thombley R, Luft HS, Sing DC, et al. Validation of Prediction Models for Critical Care Outcomes Using Natural Language Processing of Electronic Health Record Data. JAMA Netw Open. 2018 Dec 21;1(8):e185097.

5. Marafino BJ, Davies JM, Bardach NS, Dean ML, Dudley RA. N-gram Support Vector Machines for Scalable Procedure and Diagnosis Classification, with Applications to Clinical Free Text Data from the Intensive Care Unit. J Am Med Inform Assoc. 2014 Sep;21(5):871–5.

6. Marafino BJ, John Boscardin W, Adams Dudley R. Efficient and Sparse Feature Selection for Biomedical Text Classification via the Elastic Net: Application to ICU Risk Stratification from Nursing Notes. Journal of Biomedical Informatics. 2015 Apr 1;54:114–20.

7. Yann M, Stukel T, Jaakkimainen L, Tu K. Identify Patients with Congestive Heart Failure through Analyzing Free-Text Clinical Notes. International Journal of Population Data Science [Internet]. 2018 Sep 11 [cited 2023 Nov 16];3(4). Available from: https://ijpds.org/article/view/1032

8. Al Sharou K, Li Z, Specia L. Towards a Better Understanding of Noise in Natural Language Processing. In: Mitkov R, Angelova G, editors. Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP 2021) [Internet]. Held Online: INCOMA Ltd.; 2021 [cited 2023 Nov 16]. p. 53–62. Available from: https://aclanthology.org/2021.ranlp-1.7

9. Sun W, Cai Z, Li Y, Liu F, Fang S, Wang G. Data Processing and Text Mining Technologies on Electronic Medical Records: A Review. J Healthc Eng. 2018 Apr 8;2018:4302425.

10. Berndt DJ, McCart JA, Finch DK, Luther SL. A Case Study of Data Quality in Text Mining Clinical Progress Notes. ACM Trans Manage Inf Syst. 2015 Apr 3;6(1):1–21.

11. Kiefer C. Quality Indicators for Text Data [Internet]. Gesellschaft für Informatik, Bonn; 2019 [cited 2022 Mar 1]. Available from: http://dl.gi.de/handle/20.500.12116/21801

12. Kashina M, Lenivtceva ID, Kopanitsa GD. Preprocessing of Unstructured Medical Data: The Impact of Each Preprocessing Stage on Classification. Procedia Computer Science. 2020 Jan 1;178:284–90.

13. Eler DM, Grosa D, Pola I, Garcia R, Correia R, Teixeira J. Analysis of Document Pre-Processing Effects in Text and Opinion Mining. Information. 2018 Apr;9(4):100.

14. HaCohen-Kerner Y, Miller D, Yigal Y. The Influence of Preprocessing on Text Classification Using a Bag-Of-Words Representation. PLOS ONE. 2020 May 1;15(5):e0232525.

15. HaCohen-Kerner Y, Yigal Y. The Impact of Preprocessing on the Classification of Mental Disorders. 2019.

16. Naseem U, Razzak I, Eklund PW. A Survey of Pre-processing Techniques to Improve Short-text Quality: A Case Study on Hate Speech Detection on Twitter. Multimed Tools Appl. 2021 Nov 1;80(28):35239–66.

17. Mehanna YS, Mahmuddin M. The Effect of Pre-processing Techniques on the Accuracy of Sentiment Analysis Using Bag-of-Concepts Text Representation. SN COMPUT SCI. 2021 Apr 24;2(4):237.

18. Symeonidis S, Effrosynidis D, Arampatzis A. A Comparative Evaluation of Pre-processing Techniques and Their Interactions for Twitter Sentiment Analysis. Expert Systems with Applications. 2018 Nov 15;110:298–310.

19. Uysal AK, Gunal S. The Impact of Preprocessing on Text Classification. Information Processing & Management. 2014 Jan 1;50(1):104–12.

20. Kadhim A. An Evaluation of Preprocessing Techniques for Text Classification. International Journal of Computer Science and Information Security,. 2018 Jun 1;16:22–32.

21. Alnajran N, Crockett K, McLean D, Latham A. A Heuristic Based Pre-processing Methodology for Short Text Similarity Measures in Microblogs. In: 2018 IEEE 20th International Conference on High Performance Computing and Communications; IEEE 16th International Conference on Smart City; IEEE 4th International Conference on Data Science and Systems (HPCC/SmartCity/DSS). 2018. p. 1627–33.

22. Popek E. Data Quality Assessment. In: Popek E, editor. Sampling and Analysis of Environmental Chemical Pollutants (Second Edition) [Internet]. Elsevier; 2018 [cited 2023 Dec 1]. p. 303–36. Available from: https://www.sciencedirect.com/science/article/pii/B9780128032022000069

23. Cappiello C, Francalanci C, Pernici B. Data Quality Assessment from the User's Perspective. In: Proceedings of the 2004 international workshop on Information quality in information systems [Internet]. New York, NY, USA: Association for Computing Machinery; 2004 [cited 2023 Dec 1]. p. 68–73. (IQIS '04). Available from: https://doi.org/10.1145/1012453.1012465

24. Ehrlinger L, Wöß W. A Survey of Data Quality Measurement and Monitoring Tools. Frontiers in Big Data [Internet]. 2022 [cited 2023 Dec 1];5. Available from: https://www.frontiersin.org/articles/10.3389/fdata.2022.850611

25. Arts DGT, de Keizer NF, Scheffer GJ. Defining and Improving Data Quality in Medical Registries: A Literature Review, Case Study, and Generic Framework. J Am Med Inform Assoc. 2002;9(6):600–11.

26. Madnick SE, Wang RY, Lee YW, Zhu H. Overview and Framework for Data and Information Quality Research. J Data and Information Quality. 2009 Jun 1;1(1):2:1-2:22.

27. Bian J, Lyu T, Loiacono A, Viramontes TM, Lipori G, Guo Y, et al. Assessing the Practice of Data Quality Evaluation in a National Clinical Data Research Network through a Systematic Scoping Review in the Era of Real-world Data. Journal of the American Medical Informatics Association. 2020 Dec 9;27(12):1999–2010.

28. Sonntag D. Assessing the Quality of Natural Language Text Data [Internet]. Gesellschaft für Informatik e.V.; 2004 [cited 2022 Mar 1]. Available from: http://dl.gi.de/handle/20.500.12116/28866

29. Batini C, Scannapieco M. Information Quality Dimensions for Maps and Texts. In: Batini C, Scannapieco M, editors. Data and Information Quality: Dimensions, Principles and Techniques [Internet]. Cham: Springer International Publishing; 2016 [cited 2022 Oct 7]. p. 53–86. (Data-Centric Systems and Applications). Available from: https://doi.org/10.1007/978-3-319-24106-7_3

30. Kiefer C. Assessing the Quality of Unstructured Data: An Initial Overview. In: LWDA. 2016.

31. Grolemund HW and G. R for Data Science [Internet]. [cited 2023 Dec 5]. Available from: https://r4ds.had.co.nz/

32. Salah R, Mukred M, Zakaria LQ binti, Ahmed R, Sari H. A New Rule-Based Approach for Classical Arabic in Natural Language Processing. Journal of Mathematics. 2022;

33. Santaholma M. Grammar Sharing Techniques for Rule-based Multilingual NLP Systems. In: Proceedings of the 16th Nordic Conference of Computational Linguistics (NODALIDA 2007) [Internet]. Tartu, Estonia: University of Tartu, Estonia; 2007 [cited 2022 Oct 13]. p. 253–60. Available from: https://aclanthology.org/W07-2438

34. Brill E. Transformation-Based Error-Driven Learning and Natural Language Processing: A Case Study in Part-of-Speech Tagging. Computational Linguistics. 1995;21(4):543–65.

35. Wang T, Hirst G. Exploring Patterns in Dictionary Definitions for Synonym Extraction. Natural Language Engineering. 2012 Jul;18(3):313–42.

36. Assale M, Dui LG, Cina A, Seveso A, Cabitza F. The Revival of the Notes Field: Leveraging the Unstructured Content in Electronic Health Records. Frontiers in Medicine [Internet].

2019 [cited 2022 Jun 21];6. Available from: https://www.frontiersin.org/article/10.3389/fmed.2019.00066

37. Nichols JA, Herbert Chan HW, Baker MAB. Machine Learning: Applications of Artificial Intelligence to Imaging and Diagnosis. Biophys Rev. 2018 Sep 4;11(1):111–8.

38. Weissman GE, Hubbard RA, Ungar LH, Harhay MO, Greene CS, Himes BE, et al. Inclusion of Unstructured Clinical Text Improves Early Prediction of Death or Prolonged ICU Stay. Crit Care Med. 2018 Jul;46(7):1125–32.

39. Parreco J, Hidalgo A, Kozol R, Namias N, Rattan R. Predicting Mortality in the Surgical Intensive Care Unit Using Artificial Intelligence and Natural Language Processing of Physician Documentation. Am Surg. 2018 Jul 1;84(7):1190–4.

40. Sangaji AH, Pamungkas Y, Nugroho SMS, Wibawa AD. Rule-based Disease Classification using Text Mining on Symptoms Extraction from Electronic Medical Records in Indonesian. Kinetik: Game Technology, Information System, Computer Network, Computing, Electronics, and Control. 2022 Feb 28;69–80.

41. Garies S, Taylor M, Soos B, Lindeman C, Drummond N, Pham A, et al. Using Machine Learning to Standardize Medication Records in a Pan-Canadian Electronic Medical Record Database: A Data-driven Algorithm Study Focused on Antibiotics Prescribed in Primary Care. Canadian Medical Association Open Access Journal. 2023 Sep 1;11(5):E1020–4.

42. Meaney C, Moineddin R, Kalia S, Aliarzadeh B, Greiver M. Using Primary Care Clinical Text Data and Natural Language Processing to Identify Indicators of COVID-19 in Toronto, Canada. PLOS Digital Health. 2022 Dec 7;1(12):e0000150.

43. Kosowan L, Singer A, Zulkernine F, Zafari H, Nesca M, Muthumuni D. Pan-Canadian Electronic Medical Record Diagnostic and Unstructured Text Data for Capturing PTSD: Retrospective Observational Study. JMIR Medical Informatics. 2022 Dec 13;10(12):e41312.

44. Kunilovskaya M, Plum A. Text Preprocessing and its Implications in a Digital Humanities Project. In: Proceedings of the Student Research Workshop Associated with RANLP 2021 [Internet]. Online: INCOMA Ltd.; 2021 [cited 2022 Aug 12]. p. 85–93. Available from: https://aclanthology.org/2021.ranlp-srw.13

45. Dařena F, Žižka J. Interdependence of Text Mining Quality and the Input Data Preprocessing. In: Silhavy R, Senkerik R, Oplatkova ZK, Prokopova Z, Silhavy P, editors. Artificial Intelligence Perspectives and Applications. Cham: Springer International Publishing; 2015. p. 141–50.

46. About n2c2 [Internet]. [cited 2022 Jun 13]. Available from: https://n2c2.dbmi.hms.harvard.edu/about-n2c2

47. Uzuner Ö, Luo Y, Szolovits P. Evaluating the State-of-the-Art in Automatic De-identification. Journal of the American Medical Informatics Association. 2007 Sep 1;14(5):550–63.

48. Uzuner Ö, Goldstein I, Luo Y, Kohane I. Identifying Patient Smoking Status from Medical Discharge Records. Journal of the American Medical Informatics Association. 2008 Jan 1;15(1):14–24.

49. Uzuner Ö. Recognizing Obesity and Comorbidities in Sparse Data. Journal of the American Medical Informatics Association. 2009 Jul 1;16(4):561–70.

50. Lu H, Ehwerhemuepha L, Rakovski C. A Comparative Study on Deep Learning Models for Text Classification of Unstructured Medical Notes with Various Levels of Class Imbalance. BMC Med Res Methodol. 2022 Jul 2;22(1):181.

51. Su X, Miller T, Ding X, Afshar M, Dligach D. Classifying Long Clinical Documents with Pre-trained Transformers [Internet]. arXiv; 2021 [cited 2022 Dec 6]. Available from: http://arxiv.org/abs/2105.06752

52. Kumar V, Recupero DR, Riboni D, Helaoui R. Ensembling Classical Machine Learning and Deep Learning Approaches for Morbidity Identification From Clinical Notes. IEEE Access. 2021;9:7107–26.

53. Stubbs A, Kotfila C, Uzuner Ö. Automated Systems for the De-identification of Longitudinal Clinical Narratives: Overview of 2014 i2b2/UTHealth Shared Task Track 1. Journal of Biomedical Informatics. 2015 Dec 1;58:S11–9.

54. Kocaman V, Talby D. Accurate Clinical and Biomedical Named Entity Recognition at Scale. Software Impacts. 2022 Aug 1;13:100373.

55. Catelli R, Casola V, De Pietro G, Fujita H, Esposito M. Combining Contextualized Word Representation and Sub-document Level Analysis through Bi-LSTM+CRF Architecture for Clinical De-identification. Knowledge-Based Systems. 2021 Feb 15;213:106649.

56. Stubbs A, Filannino M, Soysal E, Henry S, Uzuner Ö. Cohort Selection for Clinical Trials: n2c2 2018 Shared Task Track 1. Journal of the American Medical Informatics Association. 2019 Nov 1;26(11):1163–71.

57. Henry S, Buchan K, Filannino M, Stubbs A, Uzuner O. 2018 n2c2 Shared Task on Adverse Drug Events and Medication Extraction in Electronic Health Records. Journal of the American Medical Informatics Association. 2020 Jan 1;27(1):3–12.

58. Johnson AEW, Pollard TJ, Shen L, Lehman L wei H, Feng M, Ghassemi M, et al. MIMIC-III, A Freely Accessible Critical Care Database. Sci Data. 2016 May 24;3(1):160035.

59. Alabdullah B, Beloff N, White M. ARTPHIL: Reversible De-identification of Free Text Using an Integrated Model. In: Shi W, Chen X, Choo KKR, editors. Security and Privacy in New Computing Environments. Cham: Springer International Publishing; 2022. p. 369–81. (Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering).

60. Lee K, Kayaalp M, Henry S, Uzuner Ö. A Context-Enhanced De-identification System. ACM Trans Comput Healthcare. 2021 Oct 15;3(1):6:1-6:14.

61. Murugadoss K, Rajasekharan A, Malin B, Agarwal V, Bade S, Anderson JR, et al. Building a Best-in-class Automated De-identification Tool for Electronic Health Records through Ensemble Learning. Patterns. 2021 Jun 11;2(6):100255.

62. Ahmed T, Aziz MMA, Mohammed N. De-identification of Electronic Health Record Using Neural Network. Sci Rep. 2020 Oct 29;10(1):18600.

63. Childs LC, Enelow R, Simonsen L, Heintzelman NH, Kowalski KM, Taylor RJ. Description of a Rule-based System for the i2b2 Challenge in Natural Language Processing for Clinical Data. Journal of the American Medical Informatics Association. 2009 Jul 1;16(4):571–5.

64. Ayedh A, Tan G, Alwesabi K, Rajeh H. The Effect of Preprocessing on Arabic Document Categorization. Algorithms. 2016 Jun;9(2):27.

65. Clark E, Araki K. Text Normalization in Social Media: Progress, Problems and Applications for a Pre-Processing System of Casual English. Procedia - Social and Behavioral Sciences. 2011 Jan 1;27:2–11.

66. Gharatkar S, Ingle A, Naik T, Save A. Review Preprocessing Using Data Cleaning and Stemming Technique. In: 2017 International Conference on Innovations in Information, Embedded and Communication Systems (ICIIECS). 2017. p. 1–4.

67. Gonçalves CA, Gonçalves CT, Camacho R, Oliveira E. The Impact of Pre-processing on the Classification of MEDLINE Documents. In: PRIS. 2010.

68. Haddi E, Liu X, Shi Y. The Role of Text Pre-processing in Sentiment Analysis. Procedia Computer Science. 2013 Jan 1;17:26–32.

69. Méndez JR, Iglesias EL, Fdez-Riverola F, Díaz F, Corchado JM. Tokenising, Stemming and Stopword Removal on Anti-spam Filtering Domain. In: Marín R, Onaindía E, Bugarín A, Santos J, editors. Current Topics in Artificial Intelligence [Internet]. Berlin, Heidelberg: Springer Berlin Heidelberg; 2006 [cited 2022 Jul 14]. p. 449–58. (Hutchison D, Kanade T, Kittler J, Kleinberg JM, Mattern F, Mitchell JC, et al., editors. Lecture Notes in Computer Science; vol. 4177). Available from: http://link.springer.com/10.1007/11881216_47

70. Pomikalek J. The Influence of Preprocessing Parameters on Text Categorization. 2007;4.

71. Srividhya V, Anitha R. Evaluating Preprocessing Techniques in Text Categorization. 2010;(2010):3.

72. Nesca M, Katz A, Leung C, Lix L. A Scoping Review of Preprocessing Methods for Unstructured Text Data to Assess Data Quality. International Journal of Population Data Science [Internet]. 2022 Oct 5 [cited 2022 Dec 14];7(1). Available from: https://ijpds.org/article/view/1757

73. Mujtaba G, Shuib L, Idris N, Hoo WL, Raj RG, Khowaja K, et al. Clinical Text Classification Research Trends: Systematic Literature Review and Open Issues. Expert Systems with Applications. 2019 Feb;116:494–520.

74. Sarker A, Gonzalez G. Portable Automatic Text Classification for Adverse Drug Reaction Detection via Multi-corpus Training. Journal of Biomedical Informatics. 2015 Feb 1;53:196–207.

75. Danso S, Atwell E, Johnson O. Linguistic and Statistically Derived Features for Cause of Death Prediction from Verbal Autopsy Text. In: Gurevych I, Biemann C, Zesch T, editors. Language Processing and Knowledge in the Web. Berlin, Heidelberg: Springer; 2013. p. 47–60. (Lecture Notes in Computer Science).

76. Ware H, Mullett CJ, Jagannathan V. Natural Language Processing Framework to Assess Clinical Conditions. Journal of the American Medical Informatics Association. 2009 Jul 1;16(4):585–9.

77. Mishra NK, Cummo DM, Arnzen JJ, Bonander J. A Rule-based Approach for Identifying Obesity and Its Comorbidities in Medical Discharge Summaries. Journal of the American Medical Informatics Association. 2009 Jul 1;16(4):576–9.

78. Ni Y, Bermudez M, Kennebeck S, Liddy-Hicks S, Dexheimer J. A Real-Time Automated Patient Screening System for Clinical Trials Eligibility in an Emergency Department: Design and Evaluation. JMIR Medical Informatics. 2019 Jul 24;7(3):e14185.

79. Solt I, Tikk D, Gál V, Kardkovács ZT. Semantic Classification of Diseases in Discharge Summaries Using a Context-aware Rule-based Classifier. J Am Med Inform Assoc. 2009;16(4):580–4.

80. MEDIALpy: A Small Python Package that Allows the User to Look Up Common Medical Abbreviations. [Internet]. [cited 2022 Sep 21]. Available from: https://github.com/AberystwythSystemsBiology/MEDIALpy

81. Aristotelis. glutanimate/hunspell-en-med-glut-workaround [Internet]. 2020 [cited 2022 Aug 5]. Available from: https://github.com/glutanimate/hunspell-en-med-glut-workaround

82. Bhavsar K. Stemming: Porter Vs. Snowball Vs. Lancaster – Towards AI [Internet]. [cited 2023 Dec 9]. Available from: https://towardsai.net/p/l/stemming-porter-vs-snowball-vs-lancaster, https://towardsai.net/p/l/stemming-porter-vs-snowball-vs-lancaster

83. Mohajon J. Medium. 2021 [cited 2022 Oct 6]. Confusion Matrix for Your Multi-Class Machine Learning Model. Available from: https://towardsdatascience.com/confusion-matrix-for-your-multi-class-machine-learning-model-ff9aa3bf7826

84. Subhashini R, Kumar VJS. Evaluating the Performance of Similarity Measures Used in Document Clustering and Information Retrieval. In: 2010 First International Conference on Integrated Intelligent Computing [Internet]. 2010 [cited 2023 Nov 6]. p. 27–31. Available from: https://ieeexplore.ieee.org/document/5571521

85. Jiawei Han, Jian Pei, Micheline Kamber. Data Mining: Concepts and Techniques [Internet]. Burlington, MA: Morgan Kaufmann; 2011 [cited 2023 Oct 27]. (The Morgan Kaufmann Series in Data Management Systems; vol. 3rd ed). Available from: http://uml.idm.oclc.org/login?url=https://search.ebscohost.com/login.aspx?direct=true&db=nlebk&AN=377411&site=ehost-live

86. Mujtaba G, Shuib L, Raj RG, Rajandram R, Shaikh K. Automatic Text Classification of ICD-10 Related CoD from Complex and Free Text Forensic Autopsy Reports. In: 2016 15th IEEE International Conference on Machine Learning and Applications (ICMLA). 2016. p. 1055–8.

87. García Adeva JJ, Pikatza Atxa JM, Ubeda Carrillo M, Ansuategi Zengotitabengoa E. Automatic Text Classification to Support Systematic Reviews in Medicine. Expert Systems with Applications. 2014 Mar 1;41(4, Part 1):1498–508.

88. Fernandez-Delgado M, Cernadas E, Barro S, Amorim D. Do We Need Hundreds of Classifiers to Solve Real World Classification Problems? :49.

89. Rani GJJ, Gladis D, Mammen J. Classification and Prediction of Breast Cancer Data Derived Using Natural Language Processing. In: Proceedings of the Third International Symposium on Women in Computing and Informatics [Internet]. New York, NY, USA: Association for Computing Machinery; 2015 [cited 2022 Dec 14]. p. 250–5. (WCI '15). Available from: https://doi.org/10.1145/2791405.2791489

90. Mujtaba G, Shuib L, Raj RG, Rajandram R, Shaikh K. Prediction of Cause of Death from Forensic Autopsy Reports using Text Classification Techniques: A Comparative Study. Journal of Forensic and Legal Medicine. 2018 Jul 1;57:41–50.

91. Yoon HJ, Roberts L, Tourassi G. Automated Histologic Grading from Free-text Pathology Reports using Graph-of-words Features and Machine Learning. In: 2017 IEEE EMBS International Conference on Biomedical & Health Informatics (BHI). 2017. p. 369–72.

92. Kasthurirathne SN, Dixon BE, Grannis SJ. Evaluating Methods for Identifying Cancer in Free-Text Pathology Reports Using Various Machine Learning and Data Preprocessing Approaches. MEDINFO 2015: eHealth-enabled Health. 2015;1070–1070.

93. Kasthurirathne SN, Dixon BE, Gichoya J, Xu H, Xia Y, Mamlin B, et al. Toward Better Public Health Reporting using Existing off the Shelf Approaches: A Comparison of Alternative Cancer Detection Approaches using Plain text Medical Data and Non-dictionary Based Feature Selection. Journal of Biomedical Informatics. 2016 Apr 1;60:145–52.

94. James G, Witten D, Hastie T, Tibshirani R. Tree-Based Methods. In: James G, Witten D, Hastie T, Tibshirani R, editors. An Introduction to Statistical Learning: with Applications in R [Internet]. New York, NY: Springer; 2013 [cited 2022 Oct 5]. p. 303–35. (Springer Texts in Statistics). Available from: https://doi.org/10.1007/978-1-4614-7138-7_8

95. Oleynik M, Kugic A, Kasáč Z, Kreuzthaler M. Evaluating Shallow and Deep Learning Strategies for the 2018 n2c2 Shared Task on Clinical Text Classification. Journal of the American Medical Informatics Association. 2019 Nov 1;26(11):1247–54.

96. Martinez D, Ananda-Rajah MR, Suominen H, Slavin MA, Thursky KA, Cavedon L. Automatic Detection of Patients with Invasive Fungal Disease from Free-text Computed Tomography (CT) Scans. Journal of Biomedical Informatics. 2015 Feb 1;53:251–60.

97. Cortes C, Vapnik V. Support-vector Networks. Mach Learn. 1995 Sep 1;20(3):273–97.

98. Gabriel RA, Kuo TT, McAuley J, Hsu CN. Identifying and Characterizing Highly Similar Notes in Big Clinical Note Datasets. Journal of Biomedical Informatics. 2018 Jun;82:63–9.

99. Trienes J, Trieschnigg D, Seifert C, Hiemstra D. Comparing Rule-based, Feature-based and Deep Neural Methods for De-identification of Dutch Medical Records [Internet]. arXiv; 2020 [cited 2023 Aug 3]. Available from: http://arxiv.org/abs/2001.05714

100. Khin K, Burckhardt P, Padman R. A Deep Learning Architecture for De-identification of Patient Notes: Implementation and Evaluation [Internet]. arXiv; 2018 [cited 2023 Aug 22]. Available from: http://arxiv.org/abs/1810.01570

101. Liu Z, Tang B, Wang X, Chen Q. De-identification of Clinical Notes via Recurrent Neural Network and Conditional Random Field. J Biomed Inform. 2017 Nov;75S:S34–42.

102. Yang X, Lyu T, Li Q, Lee CY, Bian J, Hogan WR, et al. A Study of Deep Learning Methods for De-identification of Clinical Notes in Cross-institute Settings. BMC Medical Informatics and Decision Making. 2019 Dec 5;19(5):232.

103. Friedrich M, Köhn A, Wiedemann G, Biemann C. Adversarial Learning of Privacy-Preserving Text Representations for De-Identification of Medical Records. In: Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics [Internet]. Florence, Italy: Association for Computational Linguistics; 2019 [cited 2023 Aug 3]. p. 5829–39. Available from: https://aclanthology.org/P19-1584

104. Huang Z, Xu W, Yu K. Bidirectional LSTM-CRF Models for Sequence Tagging [Internet]. arXiv; 2015 [cited 2023 Dec 10]. Available from: http://arxiv.org/abs/1508.01991

105. Kim HY. Statistical Notes for Clinical Researchers: Assessing Normal Distribution (2) using Skewness and Kurtosis. Restor Dent Endod. 2013 Feb;38(1):52–4.

106. Mishra P, Pandey CM, Singh U, Gupta A, Sahu C, Keshri A. Descriptive Statistics and Normality Tests for Statistical Data. Ann Card Anaesth. 2019;22(1):67–72.

107. Lin C, He Y. Joint Sentiment/Topic Model for Sentiment Analysis. In: Proceedings of the 18th ACM conference on Information and knowledge management [Internet]. New York, NY, USA: Association for Computing Machinery; 2009 [cited 2024 Jan 7]. p. 375–84. (CIKM '09). Available from: https://dl.acm.org/doi/10.1145/1645953.1646003

108. Hong N, Wen A, Stone DJ, Tsuji S, Kingsbury PR, Rasmussen LV, et al. Developing a FHIR-based EHR Phenotyping Framework: A Case Study for Identification of Patients with Obesity and Multiple Comorbidities from Discharge Summaries. Journal of Biomedical Informatics. 2019 Nov 1;99:103310.

109. Yang H, Spasic I, Keane JA, Nenadic G. A Text Mining Approach to the Prediction of Disease Status from Clinical Discharge Summaries. J Am Med Inform Assoc. 2009;16(4):596–600.

110. He B, Guan Y, Cheng J, Cen K, Hua W. CRFs Based De-identification of Medical Records. Journal of Biomedical Informatics. 2015 Dec 1;58:S39–46.

111. Dehghan A, Kovacevic A, Karystianis G, Keane JA, Nenadic G. Combining Knowledge- and Data-driven Methods for De-identification of Clinical Narratives. Journal of Biomedical Informatics. 2015 Dec 1;58:S53–9.

112. Yang H, Garibaldi JM. Automatic Detection of Protected Health Information from Clinic Narratives. Journal of Biomedical Informatics. 2015 Dec 1;58:S30–8.

113. Tannier X, Paris N, Cisneros H, Daniel C, Doutreligne M, Duclos C, et al. Hybrid Approaches for our Participation to the n2c2 Challenge on Cohort Selection for Clinical Trials [Internet]. arXiv; 2020 [cited 2023 Aug 28]. Available from: http://arxiv.org/abs/1903.07879

114. Chen L, Gu Y, Ji X, Lou C, Sun Z, Li H, et al. Clinical Trial Cohort Selection Based on Multi-level Rule-based Natural Language Processing System. Journal of the American Medical Informatics Association. 2019 Nov 1;26(11):1218–26.

115. Rawal S, Prakash A, Adhya S, Kulkarni S, Anwar S, Baral C, et al. Developing and Using Special-Purpose Lexicons for Cohort Selection from Clinical Notes [Internet]. arXiv; 2019 [cited 2023 Aug 28]. Available from: http://arxiv.org/abs/1902.09674

116. Dai HJ, Wang FD, Chen CW, Su CH, Wu CS, Jonnagaddala J. Cohort Selection for Clinical Trials using Multiple Instance Learning. Journal of Biomedical Informatics. 2020 Jul 1;107:103438.

Appendix

Table A1: Overview of preprocessing methods used by the teams/studies in the 2008, 2014 and 2018 i2b2 clinical challenges.

Teams/Studies	Preprocessing methods
Yang et al. (109)	Section and sentence splitting, Parts of Speech (POS) tagging, and shallow parsing/chunking (identify parts of speech and short phrases present in each sentence)
Solt et al. (79)	Abbreviation resolution, and splitting into different sections
Ware et al. (76)	Removal of unnecessary text from medical records. For example, removing documents that contain information on other family members
Childs et al. (63)	Splitting into sentences, tokenization, splitting into sections
Mishra et al. (77)	Removal of family history, changing question marks ("?") to the word "questionable", and changing commas (",") to periods (".")
He et al. (110)	Sentence splitting, tokenization, and removal of punctuation
Dehghan et al. (111)	tokenization, sentence splitting, POS tagging, and chunking/shallow parsing
Yang and Garibaldi (112)	sentence splitting, tokenization, POS tagging, and shallow parsing/chunking
Sorbonne Universite (113)	sentence splitting, tokenization, detecting negations and uncertainty in notes, detecting and normalizing dates, spelling correction, and splitting section
Med Data Quest (114)	Sentence division and segmentation, tokenization, section detection, and spelling correction
Cincinnati Children's Hospital Medical Center (78)	Tokenization, lemmatization, removing duplicate sentences, punctuation and stop words
Arizona State University (115)	Tokenization, sentence segmentation, section identification, detecting negations
National Taitung, Taipei Medical, University of New South Wales (116)	Tokenization, identifying section

Table A2: Mean estimates (SD) of evaluation metrics with and without removing stop words for Objective 1 (detecting health conditions) that had six orders with removing stop words and two orders with not removing stop words.

	Order	Sensitivity (SD)	Specificity (SD)	F1 Score (SD)	Accuracy (SD)	Precision (SD)
Removing stop words	1	0.77 (0.08)	0.79 (0.08)	0.61 (0.21)	0.78 (0.08)	0.55 (0.28)
	2	0.77 (0.08)	0.79 (0.08)	0.61 (0.21)	0.78 (0.08)	0.55 (0.28)
	3	0.76 (0.08)	0.79 (0.08)	0.61 (0.21)	0.78 (0.08)	0.55 (0.28)
	4	0.77 (0.07)	0.79 (0.08)	0.61 (0.21)	0.78 (0.08)	0.55 (0.28)
	5	0.77 (0.08)	0.79 (0.08)	0.60 (0.22)	0.78 (0.07)	0.55 (0.28)
	6	0.77 (0.08)	0.79 (0.08)	0.61 (0.21)	0.78 (0.07)	0.55 (0.28)
Not removing stop words	1	0.76 (0.08)	0.78 (0.08)	0.60 (0.21)	0.77 (0.08)	0.54 (0.28)
	2	0.76 (0.07)	0.78 (0.08)	0.60 (0.21)	0.77 (0.08)	0.54 (0.28)

Note: SD = Standard Deviation

Table A3: Mean estimates (SD) of evaluation metrics with and without removing stop words for Objective 2 (identifying selection criteria) that had six orders with removing stop words and two orders with not removing stop words.

	Order	Sensitivity (SD)	Specificity (SD)	F1 Score (SD)	Accuracy (SD)	Precision (SD)
Removing stop words	1	0.50 (0.40)	0.59 (0.38)	0.49 (0.37)	0.75 (0.19)	0.50 (0.35)
	2	0.50 (0.40)	0.59 (0.38)	0.49 (0.37)	0.75 (0.19)	0.50 (0.35)
	3	0.50 (0.40)	0.59 (0.38)	0.49 (0.37)	0.75 (0.20)	0.50 (0.35)
	4	0.50 (0.40)	0.59 (0.38)	0.49 (0.37)	0.75 (0.19)	0.50 (0.35)
	5	0.50 (0.40)	0.59 (0.38)	0.49 (0.37)	0.75 (0.19)	0.50 (0.35)
	6	0.50 (0.40)	0.58 (0.38)	0.49 (0.37)	0.75 (0.20)	0.50 (0.35)
Not removing stop words	1	0.50 (0.38)	0.59 (0.38)	0.49 (0.36)	0.75 (0.20)	0.50 (0.35)
	2	0.49 (0.39)	0.59 (0.37)	0.49 (0.37)	0.75 (0.20)	0.49 (0.35)

Note: SD = Standard Deviation

Table A4: Mean estimates (SD) of cosine similarity scores with and without removing stop words for Objective 3 (similarity measure) that had six orders with removing stop words and two orders with not removing stop words.

	Order	Mean (SD)
Removing stop words	1	0.24 (0.15)
	2	0.24 (0.16)
	3	0.24 (0.15)
	4	0.24 (0.15)
	5	0.24 (0.15)
	6	0.24 (0.15)
Not removing stop words	1	0.49 (0.17)
	2	0.49 (0.17)

Note: SD = Standard Deviation

Table A5: Mean estimates (SD) of evaluation metrics with and without removing unnecessary information to preprocess the 2008 i2b2 dataset for Objective 1 (detecting health conditions).

	Order	Sensitivity (SD)	Specificity (SD)	F1 Score (SD)	Accuracy (SD)	Precision (SD)
Removing unnecessary information	1	0.76 (0.08)	0.78 (0.08)	0.60 (0.21)	0.77 (0.08)	0.54 (0.28)
	2	0.76 (0.07)	0.78 (0.08)	0.60 (0.21)	0.77 (0.08)	0.54 (0.28)
Not removing unnecessary information	1	0.75 (0.08)	0.77 (0.09)	0.59 (0.22)	0.76 (0.08)	0.53 (0.28)
	2	0.76 (0.07)	0.77 (0.08)	0.59 (0.21)	0.76 (0.08)	0.53 (0.28)

Note: SD = Standard Deviation; Unnecessary information refers to individual names, dates, phone/contact numbers, batch number and documentation code

Table A6: Mean estimates (SD) of evaluation metrics with and without removing unnecessary information to preprocess the 2018 i2b2 dataset for Objective 2 (identifying selection criteria).

	Order	Sensitivity (SD)	Specificity (SD)	F1 Score (SD)	Accuracy (SD)	Precision (SD)
Removing unnecessary information	1	0.50 (0.38)	0.59 (0.38)	0.49 (0.36)	0.75 (0.20)	0.50 (0.35)
	2	0.49 (0.39)	0.59 (0.37)	0.49 (0.37)	0.75 (0.20)	0.49 (0.35)
Not removing unnecessary information	1	0.50 (0.39)	0.59 (37)	0.49 (0.37)	0.75 (0.20)	0.50 (0.35)
	2	0.50 (0.39)	0.58 (38)	0.49 (0.37)	0.75 (0.20)	0.50 (0.35)

Note: SD = Standard Deviation; Unnecessary information refers to individual names, dates, phone/contact numbers, batch number and documentation code

Table A7: Mean estimates (SD) of cosine similarity scores with and without removing unnecessary information to preprocess the 2018 i2b2 dataset of paired notes for Objective 3 (measuring similarity of information).

Order	Removing unnecessary information	Not removing unnecessary information
1	0.49 (0.17)	0.48 (0.17)
2	0.49 (0.17)	0.48 (0.17)

Note: SD = Standard Deviation; Unnecessary information refers to individual names, dates, phone/contact numbers, batch number and documentation code

Table A8: Comparison of evaluation metrics among different oversampling and under sampling techniques used for random forest (RF) model when using four preprocessing methods for order 1 in the 2008 i2b2 dataset.

RF model with over/under sampling techniques	Sensitivity (SD)	Specificity (SD)	F1 Score (SD)	Accuracy (SD)	Precision (SD)
Balanced RF model	0.76 (0.08)	0.78 (0.08)	0.60 (0.21)	0.77 (0.08)	0.54 (0.28)
RF model with balanced class weight	0.34 (0.43)	0.86 (0.26)	0.34 (0.40)	0.85 (0.05)	0.53 (0.37)
Random oversampling	0.39 (0.39)	0.88 (0.22)	0.42 (0.36)	0.86 (0.04)	0.74 (0.36)
SMOTE oversampling	0.40 (0.38)	0.88 (0.24)	0.43 (0.34)	0.86 (0.04)	0.78 (0.26)
Combination of SMOTE and under sampling	0.40 (0.39)	0.88 (0.24)	0.43 (0.35)	0.86 (0.05)	0.75 (0.27)
SMOTE Tomek technique (Combination of SMOTE and Tomek under sampling)	0.40 (0.39)	0.88 (0.26)	0.43 (0.40)	0.86 (0.05)	0.75 (0.27)

Note: SD = Standard Deviation; SMOTE stands for Synthetic Minority Oversampling Technique and oversamples the minority class and under samples the majority class; SMOTE-Tomek technique is the combination of SMOTE (over sampling) and Tomek (under sampling).

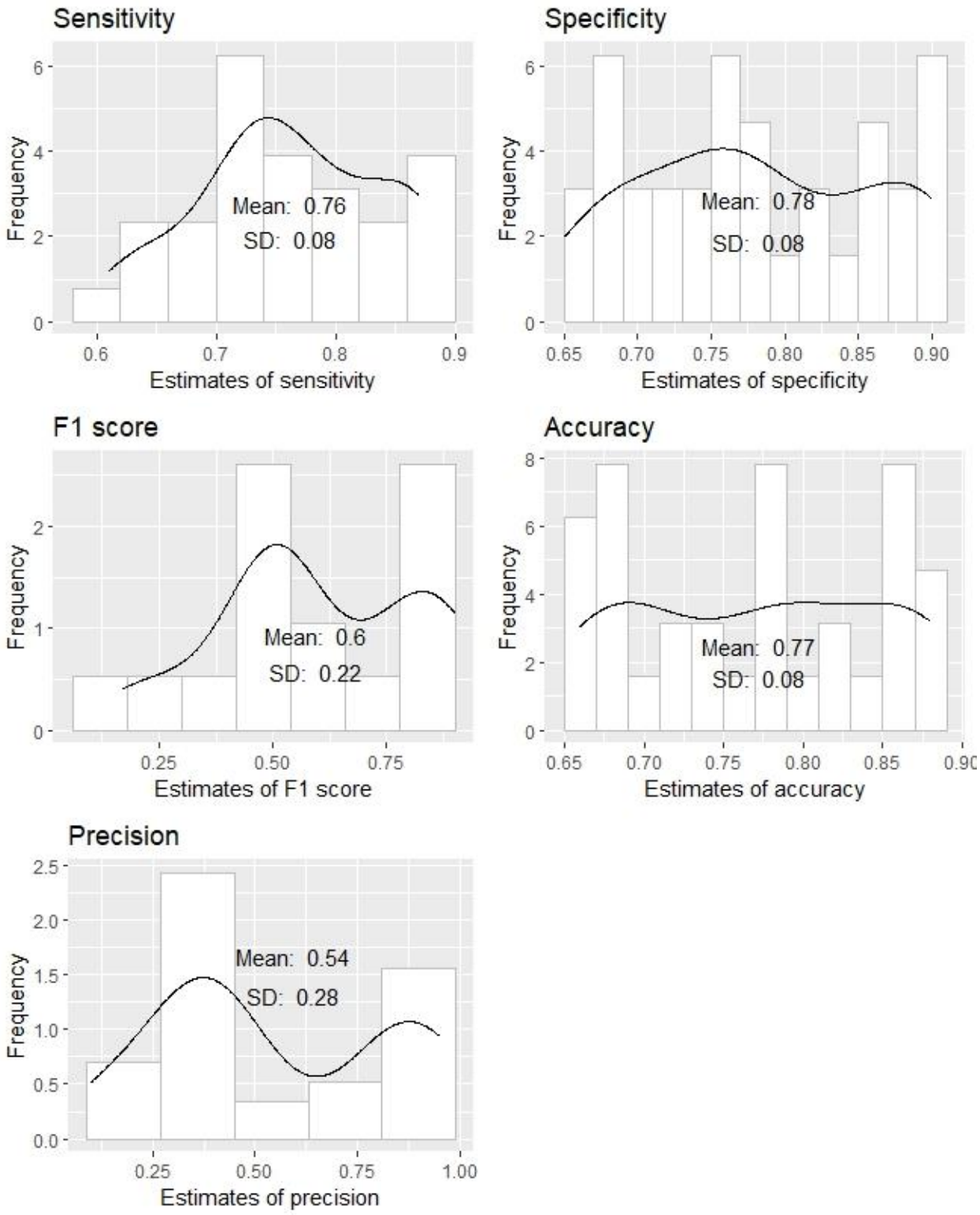


Figure A1: Histograms of evaluation metrics for four preprocessing methods applied to the 2008 i2b2 dataset for Objective 1 (detection of health conditions).

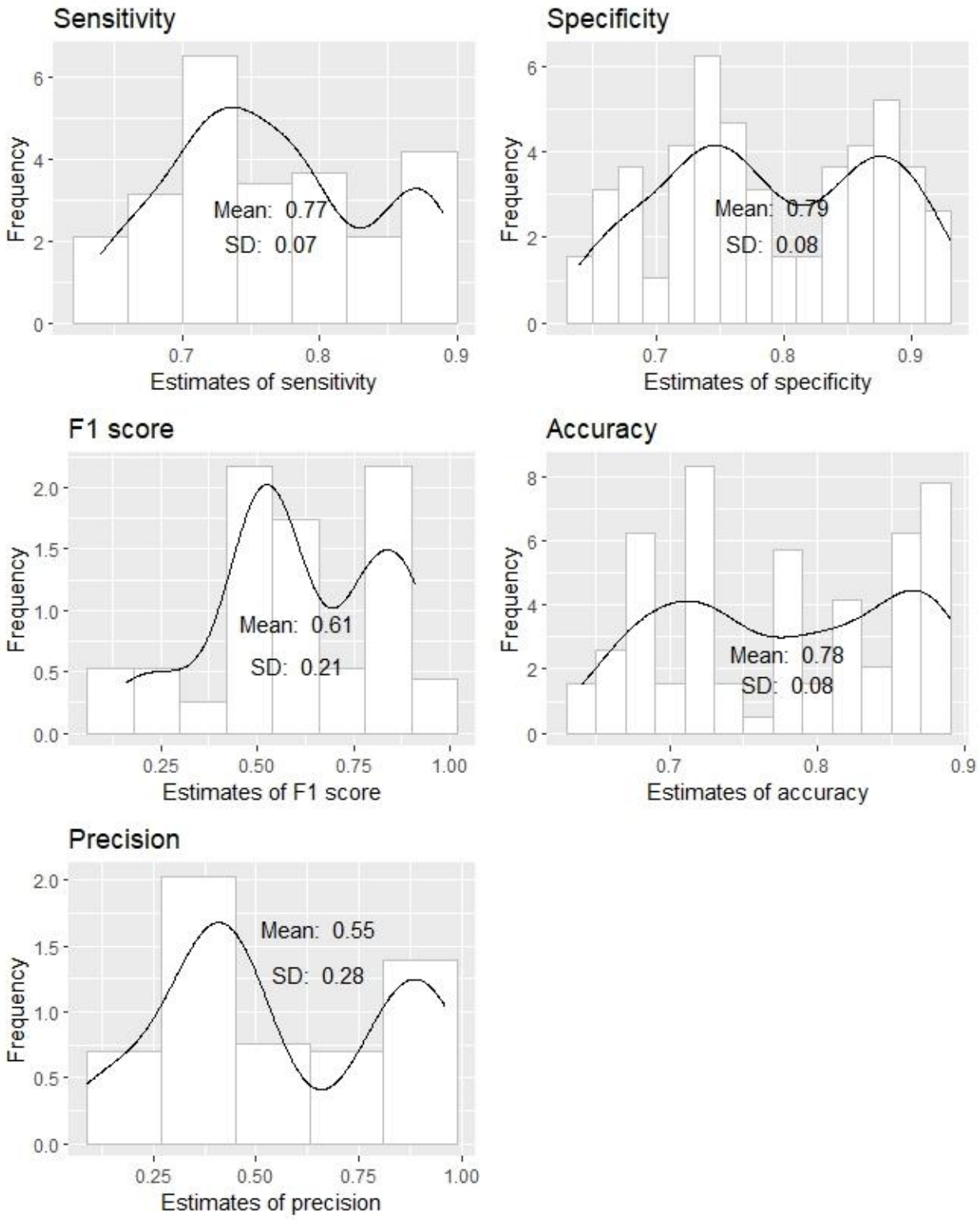


Figure A2: Histograms of evaluation metrics for five preprocessing methods including stemming applied to the 2008 i2b2 dataset for Objective 1 (detection of health conditions).

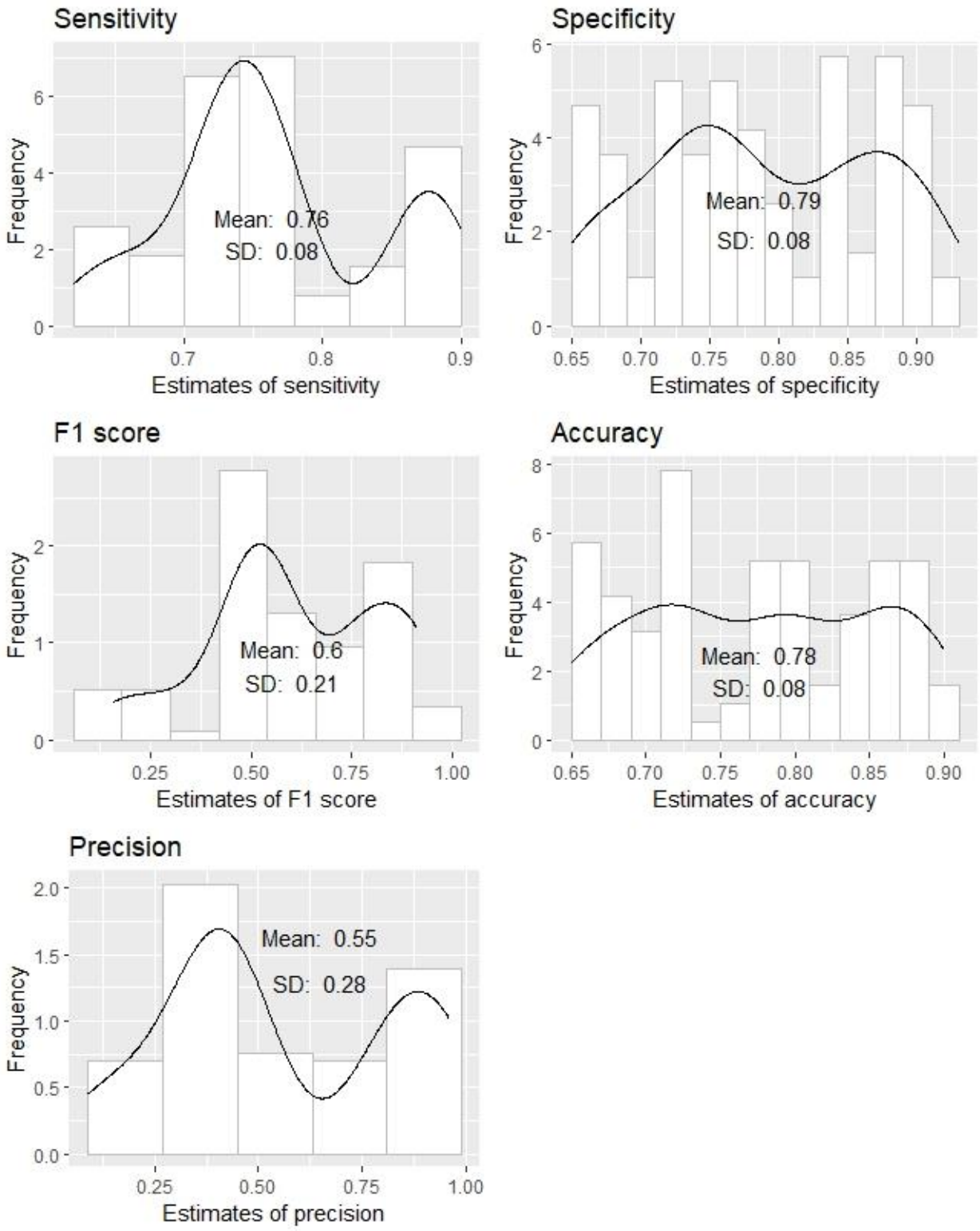


Figure A3: Histograms of evaluation metrics for five preprocessing methods including lemmatization applied to the 2008 i2b2 dataset for Objective 1 (detection of health conditions).

Table A9: The mean, median, skewness, and kurtosis of evaluation metrics for preprocessing methods applied to the 2008 i2b2 dataset for Objective 1 (detection of health conditions).

Number of preprocessing methods	Evaluation metrics	Mean	Median	Skewness	Kurtosis
4	Sensitivity	0.76	0.76	-0.15	2.14
	Specificity	0.78	0.78	0.06	1.75
	F1 Score	0.60	0.54	-0.16	2.12
	Accuracy	0.77	0.79	-0.03	1.60
	Precision	0.54	0.42	0.20	1.60
5 (with stemming)	Sensitivity	0.77	0.76	0.21	2.02
	Specificity	0.79	0.78	-0.04	1.77
	F1 Score	0.61	0.57	-0.26	2.27
	Accuracy	0.78	0.78	-0.07	1.59
	Precision	0.55	0.47	0.15	1.69
5 (with lemmatization)	Sensitivity	0.76	0.75	0.29	2.28
	Specificity	0.79	0.78	<-0.01	1.82
	F1 Score	0.60	0.57	-0.23	2.30
	Accuracy	0.78	0.78	-0.03	1.73
	Precision	0.55	0.47	0.17	1.69

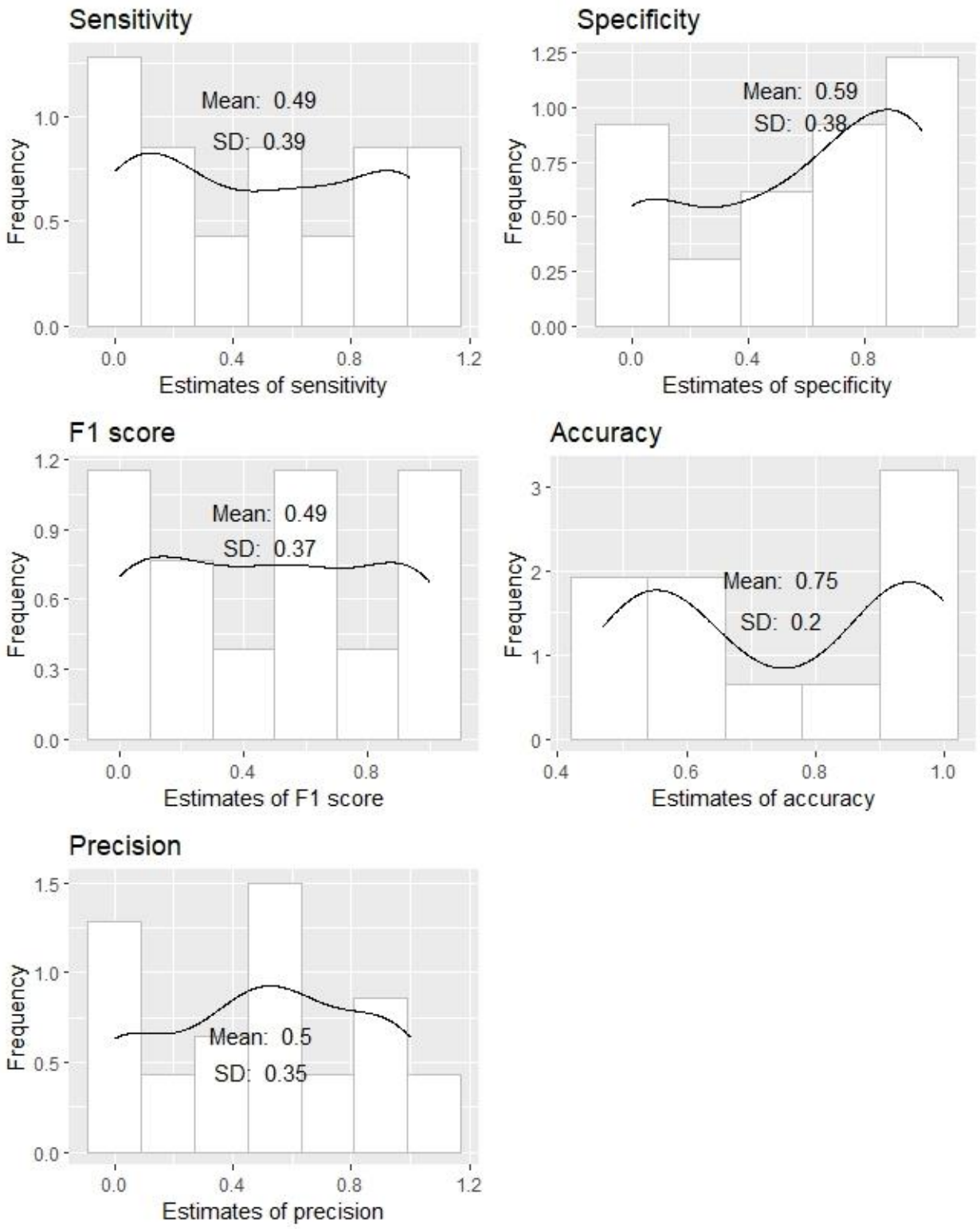


Figure A4: Histograms of evaluation metrics for four preprocessing methods applied to the 2018 i2b2 dataset for Objective 2 (identification of cohort selection criteria).

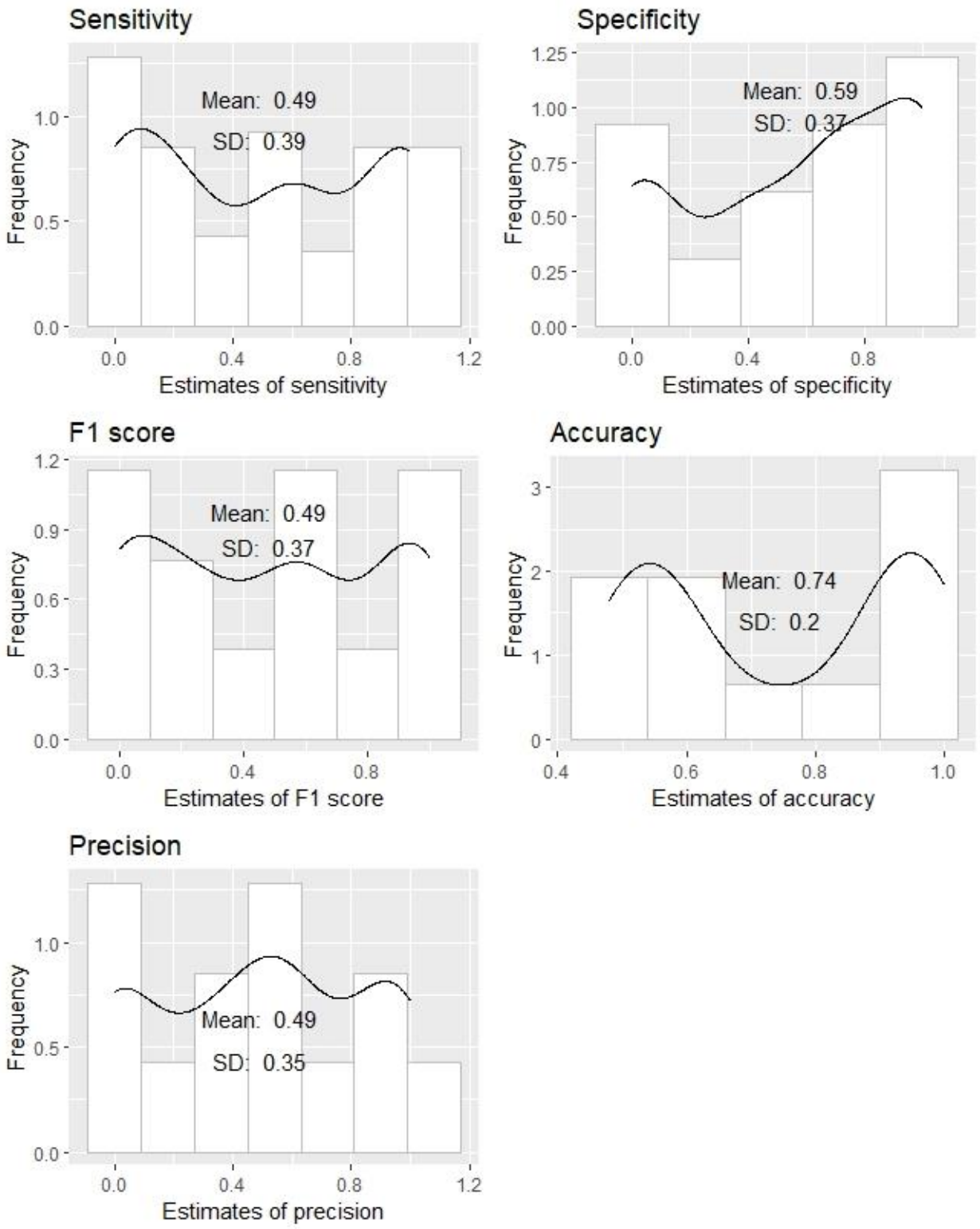


Figure A5: Histograms of evaluation metrics for five preprocessing methods including stemming applied to the 2018 i2b2 dataset for Objective 2 (identification of cohort selection criteria).

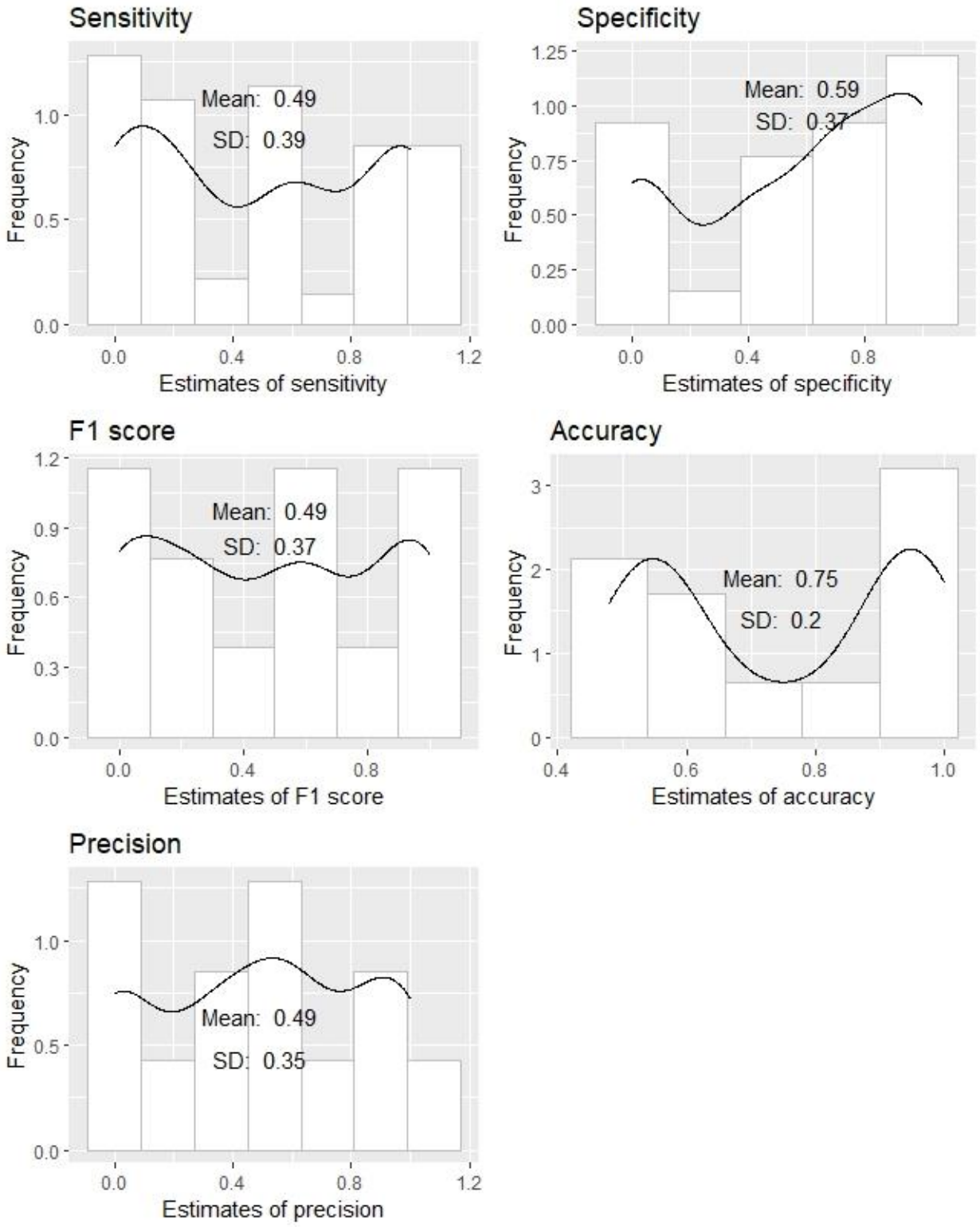


Figure A6: Histograms of evaluation metrics for five preprocessing methods including lemmatization applied to the 2018 i2b2 dataset for Objective 2 (identification of cohort selection criteria).

Table A10: The mean, median, skewness, and kurtosis of evaluation metrics for preprocessing methods applied to the 2018 i2b2 dataset for Objective 2 (identification of cohort selection criteria).

Number of preprocessing methods	Evaluation metrics	Mean	Median	Skewness	Kurtosis
4	Sensitivity	0.49	0.56	0.06	1.48
	Specificity	0.59	0.73	-0.40	1.70
	F1 Score	0.49	0.54	0.01	1.58
	Accuracy	0.75	0.75	-0.04	1.25
	Precision	0.50	0.51	-0.11	1.77
5 (with stemming)	Sensitivity	0.49	0.55	0.07	1.46
	Specificity	0.59	0.70	-0.38	1.72
	F1 Score	0.49	0.56	0.03	1.55
	Accuracy	0.74	0.75	-0.03	1.23
	Precision	0.49	0.52	-0.05	1.71
5 (with lemmatization)	Sensitivity	0.49	0.56	0.07	1.46
	Specificity	0.59	0.71	-0.42	1.75
	F1 Score	0.49	0.54	0.03	1.55
	Accuracy	0.75	0.75	-0.03	1.23
	Precision	0.49	0.52	-0.07	1.72

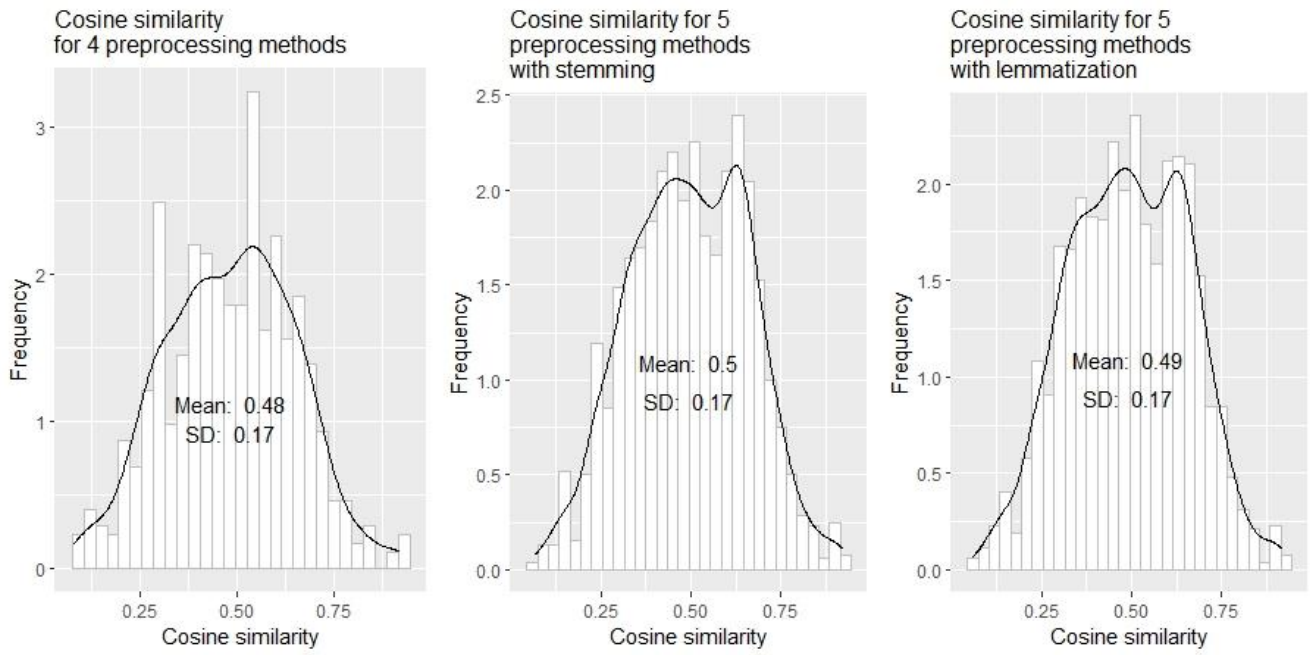


Figure A7: Histograms of cosine similarity for preprocessing methods applied to the subset of the 2018 i2b2 dataset of paired notes of the same patient for Objective 3 (similarity measures).

Table A11: The mean, median, skewness, and kurtosis of evaluation metrics for preprocessing methods applied to the subset of the 2018 i2b2 dataset of paired notes of same patient for Objective 3 (similarity measures).

Number of preprocessing methods	**Mean**	**Median**	**Skewness**	**Kurtosis**
4	0.49	0.49	-0.02	2.62
5 (with stemming)	0.50	0.50	-0.04	2.49
5 (with lemmatization)	0.49	0.50	-0.03	2.46