BENCHMARKING THE PERFORMANCE OF MACHINE LEARNING ALGORITHMS FOR RECORD LINKAGE AT DIFFERENT HETEROGENEITY RATES IN A NEW SETTING

An Undergraduate Research Scholars Thesis

by

HARIHARAN SIVAKUMAR

Submitted to the LAUNCH: Undergraduate Research office at
Texas A&M University
in partial fulfillment of requirements for the designation as an

UNDERGRADUATE RESEARCH SCHOLAR

Approved by
Faculty Research Advisor: Dr. Hye-Chung Kum

May 2022

Major: Computer Engineering – Computer Science

RESEARCH COMPLIANCE CERTIFICATION

Research activities involving the use of human subjects, vertebrate animals, and/or biohazards must be reviewed and approved by the appropriate Texas A&M University regulatory research committee (i.e., IRB, IACUC, IBC) before the activity can commence. This requirement applies to activities conducted at Texas A&M and to activities conducted at non-Texas A&M facilities or institutions. In both cases, students are responsible for working with the relevant Texas A&M research compliance program to ensure and document that all Texas A&M compliance obligations are met before the study begins.

I, Hariharan Sivakumar, certify that all research compliance requirements related to this Undergraduate Research Scholars thesis have been addressed with my Research Faculty Advisor prior to the collection of any data used in this final thesis submission.

This project did not require approval from the Texas A&M University Research Compliance & Biosafety office.

TABLE OF CONTENTS

ABSTRACT

Benchmarking the Performance of Machine Learning Algorithms for Record Linkage at Different Heterogeneity Rates in a New Setting

Hariharan Sivakumar
Department of Computer Science and Engineering
Texas A&M University

Research Faculty Advisor: Dr. Hye-Chung Kum
Department of Computer Science and Engineering
Texas A&M University

Record linkage is used to identify and link the same entity from one or more databases when a unique identifier is absent. As the amount of data increases largely every day, machine learning has become effective in integrating data with heterogeneity from multiple sources to establish more comprehensive datasets. As it is challenging to build a high-quality labeled dataset to train good models, our aim for this research will be to investigate which machine learning models will work best under certain conditions when applying these models trained in one setting to a new setting. In this paper, we compare the performance of three different machine learning models (i.e., random forests, linear SVM, and radial SVM) trained in a different setting from an open-source hybrid record linkage system using different heterogeneity rates (0% - 60%). The RL heterogeneity generator introduces name errors, date errors, missing data errors, and record level heterogeneities in the data. The models were trained on a subset of hospital record data containing nearly 10,000 pairs. We test how robust these models are in a new voter registration dataset. The performance of the models was evaluated based on F1 score,

Recall, and the percentage of pairs that needed manual review. The radial and linear SVM models transfer better to a new setting across all heterogeneity rates compared to the random forest model. The linear SVM model outperformed the radial SVM by 4% on average in terms of the percentage of pairs that needed manual review. However, we found that the radial SVM performed significantly better than the linear SVM in terms of recall performance (80% - 48% compared to 59% - 29%) for heterogeneity rates from 0% to 60%. Overall, the radial SVM performed best in our experiments.

DEDICATION

To my friends, families, instructors, and peers who supported me throughout the research process.

ACKNOWLEDGEMENTS

Contributors

I would like to thank and acknowledge my faculty advisor, Dr. Hye-Chung Kum, whose knowledge, experience, and support has made this research work possible.

Thanks also go to my friends and colleagues and the Department of Computer Science and Engineering faculty and staff for making my time at Texas A&M University a great experience.

I would also like to thank a member of the lab, Mahin Ramezani, for taking the time out of her schedule for contributing and providing me with insightful feedback. The heterogeneity generator code used for the heterogeneity generation in this research was provided by another member of the lab, Gurudev Ilangovan. All other work conducted for the thesis was completed by the student independently.

Funding Sources

This work was conducted with support in part by the Population Informatics Lab and the Texas Virtual Data Library (ViDaL) at Texas A&M University. Any opinions, findings, and conclusions or recommendations expressed in the project material are those of the authors and do not necessarily reflect the views of the funders.

NOMENCLATURE

EHR	Electronic Health Record
ML	Machine Learning
RF	Random Forest
SVM	Support Vector Machine

1. INTRODUCTION

As the amount of data increases largely each day, it is essential to combine the data from different sources in order to have sufficient data for many analyses. The main motivation for this paper begins with the importance of record linkage in our day to day lives. For example, many hospitals and research centers contain records of COVID-19 tests of patients. However, they contain different but complementing data about the same patients. Hence, linking these different records/databases, allows us to analyze the data to better comprehend death rates and understand COVID-19 health service usage and other drugs and therapies. With these data resources combined, it allows researchers to accelerate the study of the disease pathway for patients with COVID-19 from primary to critical care.

Record linkage refers to linking records or entities across various databases that have no unique identifier (Ilangovan, 2019). The main challenge that is faced when observing data from databases is heterogeneity, meaning the records can be different or have different attributes for a variety of reasons. Error-free databases that have common unique identifiers can be easily integrated with simple joins but these types of identifiers are not available in real world datasets. The type of heterogeneities includes name heterogeneities, date errors, missing data errors, and record level heterogeneities (e.g., duplicate records). These heterogeneities and errors will be introduced into the dataset at different rates to analyze how robust each model is.

Automated record linkage has been analyzed in many areas by researchers, but this research will consist of using a hybrid record linkage framework which will combine the manual review and the automated process to produce high quality results (Ramezani, 2021). The manual review process involves human experts that review uncertain pairs which are created by the

algorithms and then make decisions and conclude. The most frequently used approaches were probabilistic methods; however, machine learning models are now shown to perform better for automatic linkage methods (Ramezani, 2021). The main goal of the hybrid record linkage system is to achieve the highest linkage quality while decreasing the manual review required.

Since developing high quality labeled data is very challenging, it is important to understand how machine learning models, trained in one setting can perform on another one. To answer this question, we evaluated the performance of different RL trained machine learning models on a new dataset. In addition to analyzing performance of models in a new setting, we introduced different rates of heterogeneity into our dataset to examine how well these RL models perform under diverse conditions.

In this research, by adding different rates of heterogeneity to the data, we compare our machine learning models, trained on a subset of hospital record data containing 10,000 pairs, to understand how robust these models are in a new setting. In summary, this paper will contribute to evaluating the robustness of three different machine learning models trained in a one setting to data in a new setting with different heterogeneity rates.

1.1 Overview of Record Linkage

Machine learning algorithms are tuned to minimize false matches which leads to incorrectly identifying true matches causing datasets to be fragmented (Ramezani, 2021). In contrast, manual record linkage methods have become tedious and labor intensive. This study will consist of using a hybrid record linkage framework which uses both the automatic record linkage process as well as the manual review process. The automatic record linkage process consists of three main components: pair generation, feature extraction, and paired comparison through the ML models. As seen in **Figure 1.1**, once the two datasets are preprocessed and cleaned, the automatic record linkage process begins by combining the two datasets and generating pairs. After pair generation, features are extracted from the pairs and each machine learning model compares the pairs and classifies them into three classes, match, unmatched, and uncertain. The automated record linkage process will take care of the records that are either match or unmatched, which for most scenarios is majority of the data. However, the manual review process occurs on the pairs the algorithm was uncertain about where each pair is manually reviewed and matched as appropriate.

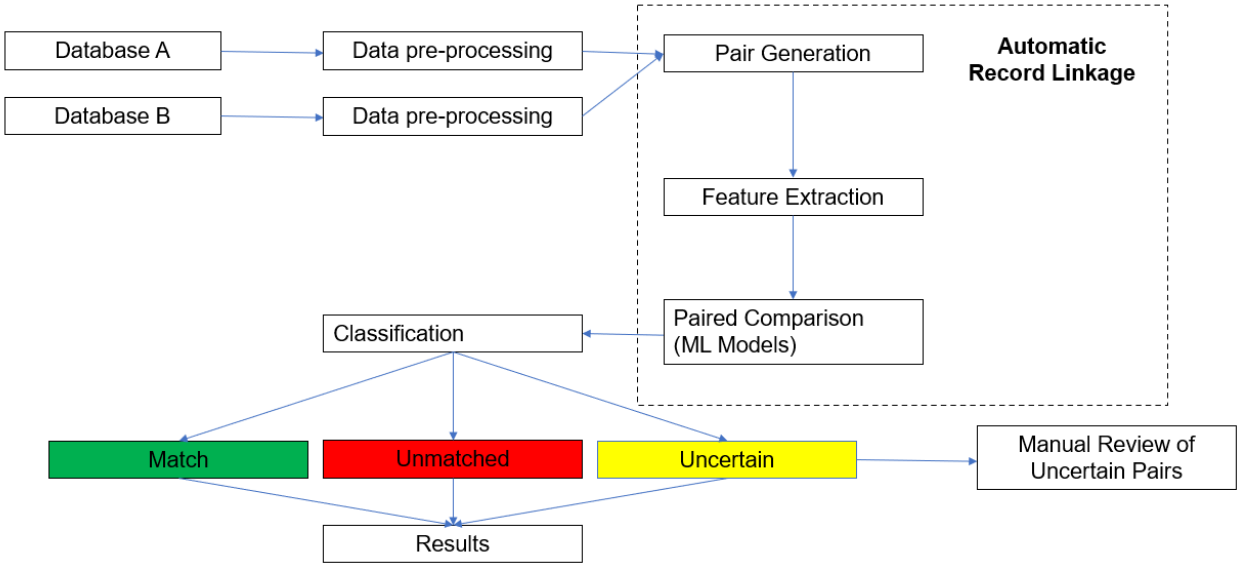

Figure 1.1: Record Linkage Process

1.2 Related Works

These studies are related to the work in this paper and compare the performance of different machine learning algorithms in different settings. This section describes the different approaches and results of the two studies.

1.2.1 Ilangovan et al.

In Ilangovan's work, he studied the effectiveness and efficiency of different ML algorithms (SVM, Random Forest, and neural networks) with different levels of heterogeneity (0% to 60% in steps of 5%) and different sizes of training dataset (Ilangovan, 2019). Each model was evaluated with the F1 score and the percentage of manual review. It was found that the random forest model and the SVM performed very well in terms of F1 score and manual review for heterogeneity rates from 0% to 60%.

1.2.2 Ramezani et al.

Ramezani studied the performance of four different models (Random Forest, Linear SVM, Radial SVM, and Dense Neural Networks) in different settings based on F1 score, recall, and number of pairs that needed manual review (Ramezani, 2021). A hybrid record linkage framework was also used in this paper. It was found that the RF, linear SVM and radial SVM models transfer to a new setting better compared to the Dense Neural Network. She also studied the effect of name2vec feature on each model's performance. It was found that using n2v results in a slightly lower F1 score. Overall, the SVM models performed best in all experiments.

2. METHODS

2.1 Data

For this study, three machine learning models, trained on EHR hospital data (Ramezani, 2021), were evaluated on the North Carolina voter registry dataset, which is publicly available, to understand how they perform in a different setting. In addition, different rates of heterogeneity were added to NC voter registry dataset (Ilangovan, 2019) to compare how robust those models are.

North Carolina voter registry data (NCSBE, 2021) contains up-to-date information for individuals who are registered or formally registered to vote in North Carolina. However, voter birthdate, social security number, and driver's license numbers are not included as it is confidential under state law. Two different time points of dataset from May 2017 and July 2020 were utilized for this experiment with the voter registry number being used as the gold standard. For this experiment, there were a total of 10,000 pairs which were sampled from Yancey County. As the NC voter registry data only contains birth year, a real dataset containing a distribution of date of birth in the US has been used to generate synthetic full dates of birth to NC dataset (Ilangovan, 2019).

2.2 Pair File Generation

The first step of the hybrid record linkage process is to generate pairs from the datasets being linked for potential matches. Often known as multipass blocking, the identifier field is utilized to generate potential pairs. In multipass blocking, the pairs are generated by blocking on certain fields where a block consists of all records that have the same value for the fields and paired together. It is called multipass, because typically this is done more than once to ensure

that matches that are not identical in one pass, will be paired in another pass as indicated in **Figure 2.1** (Ramezani, 2021).

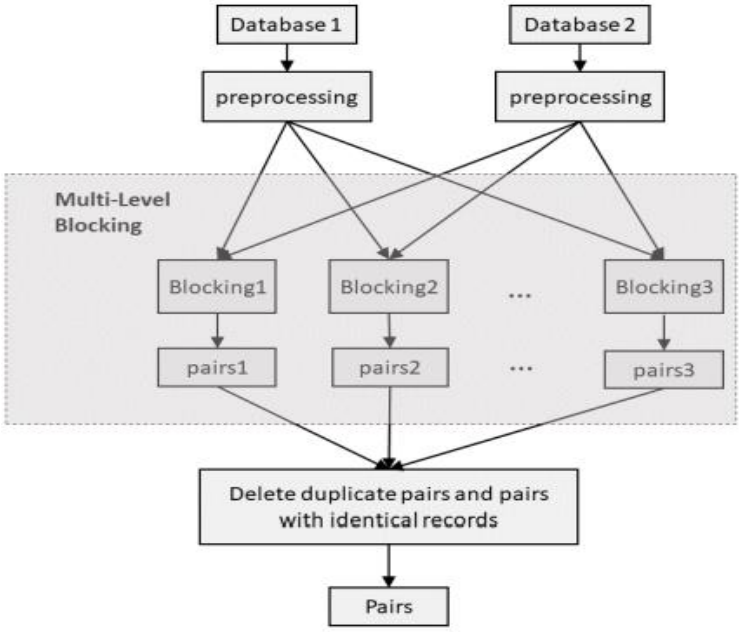

Figure 2.1: Generating pairs from two datasets

For example, blocking can be done on the fields, first name and last name, first name and date of birth, and last name and date of birth. An example of the blocking done on the fields, first name and date of birth, can be seen in **Table 2.1**. As the first name and date of birth fields for the two records have the same value, the fields for each record are blocked and the pair is generated. Once the pairs are created from the datasets being linked, errors will be introduced into the pair files at different rates and features would be extracted from each pair and fed into the ML models.

Table 2.1: Example of Records Blocked on Fname and DOB

Fname	Lname	DOB	Sex	Race
John	Miller	2000-07-29	M	B
John	Anderson	2000-07-29	M	B

2.3 Heterogeneity Generation

The type of heterogeneities introduced were name heterogeneities, date errors, missing data errors, and record level heterogeneities (e.g. duplicate records) (Ilangovan, 2019). A benchmarking system was used (Ilangovan, 2019) to introduce heterogeneities into the dataset at different rates from 0% to 60% in steps of 5%. Using the heterogeneity generator, the type of heterogeneity, the field that we want it in, and the amount of heterogeneity can be controlled. This system takes in two parameters: the rate, and the heterogeneity distribution, which is the proportion of the different heterogeneities. The generator will then introduce heterogeneities until the specified percentage of records contain the errors. As seen in **Table 2.2**, the different types of heterogeneity introduced in the data are listed along with examples of each type of heterogeneity.

Table 2.2: Name Heterogeneities and Date Errors introduced into dataset

Name Heterogeneities	**Date Errors**
Typographical errors E.g.,: Indel – John vs Jon Replace- Cristen vs Kristen Transpose – Johnathan vs Johanthan	Day – Month Swaps E.g.,: 1952-01-06 vs 1952 -06-01
Switching first names with nicknames E.g., – Switching Abigail with Abby	Month Replace E.g.,: 1952-01-06 to 1952-09-06
Suffix additions (e.g., “JR”, “SR”, “I”, etc.) Random males were chosen, and a suffix was added to their last names.	Day Replace E.g.,: 1952-01-06 to 1952-01-13
Marital name changes E.g., – Females over 20 were selected and their last names were randomly changed.	Day Transpose E.g., - 03/29/1994 vs 03/92/1994 (invalid) - 04/02/1998 vs 04/20/1992 (valid)
First name and last name swaps E.g., Mary Jane vs Jane Mary	Year Replace E.g.,: 1952-01-06 to 1969-01-06
	Year Transpose E.g., 08/10/1967 vs 08/10/1976

Missing data errors were also introduced where a field for a record would be removed. A field would be selected at random and be erased. For example, if the attribute last name was selected to be made missing, a few records would be selected at random, and their last names would be erased. These heterogeneities will be introduced into the new setting (Voter Registry Data) and the ML models will be assessed on how it performs under different rates of heterogeneity.

2.4 Feature Extraction Design

Once pairs are generated, it is time to extract features and then feed them into the three ML algorithms. The same features which were used in the training steps is needed (Ramezani, 2021). In the following we introduce those features and the similarity distances (Jaro Winkler distance, Damerau-Levenshten distance, Longest Common Subsequence distance, Soundex distance) used to extract features from pairs. The distance between two strings ranges from 0 to 1 where 1 means string are equal and 0 means no similarity (Ramezani, 2021). The main features that were extracted were name features, date features, and other features such as address, gender, phone number, and voter registry number.

For each pair of first and last names, the Jaro Winkler (JW), Damerau-Levenshtein (DL), Longest Common Subsequence (LCS), and Soundex distances. For each date of birth pairs, the Damerau-Levenshtein distances (DL) for the year, month, and day components were calculated individually. For each pair of addresses, genders, phone numbers and voter registry numbers, the Damerau-Levenshtein (DL) and Longest Common Subsequence distances were calculated. Once the similarity distances are used to extract features from the pairs with heterogeneity, the three algorithms are run on these pair files and assessed.

2.5 Evaluation

For evaluating each model, three sets of criteria was used: F1 Score, Recall, and the number of pairs that needed manual review.

2.5.1 F1 Score, Recall, and Confusion Matrix

For this experiment, the evaluation methods used are the F1 score ($F1_{auto}$) and recall, which consists of lower bound recall ($recall_{lower}$) and upper bound recall ($recall_{upper}$). The F1 score ($F1_{auto}$) is calculated only for the pairs that are classified automatically. This will help us understand how well the automatic component of the hybrid record linkage system has performed. In addition to the $F1_{auto}$, the lower bound and upper bound recall will be calculated to understand the limits of the performance of each model in the record linkage system. The $recall_{lower}$ measures the minimum performance of the model when there is no manual review done and all uncertain pairs are considered a nonmatch. In comparison, the $recall_{upper}$ measures the maximum performance of the model when manual review is done with the assumption that during the manual review all matches were correctly identified.

The confusion matrix in **Table 2.3** is a 2x2 matrix that classifies the matched/unmatched predicted and actual values into TP (True Positive), FN (False Negative), FP (False Positive), and TN (True Negative).

Table 2.3: Confusion Matrix

		Predicted	
		Match	Unmatched
Actual	Match	TP	FN
	Unmatched	FP	TN

Using the four values (TP, TN, FP, FN) from the confusion matrix, the formula for F1 score ($F1_{auto}$) can be derived as follows.

$$Precision = \frac{TP}{TP+FP} \quad (2.1)$$

$$Recall = \frac{TP}{TP+FN} \quad (2.2)$$

$$F1\text{-}Score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (2.3)$$

The formulas for $recall_{lower}$ and $recall_{upper}$ are derived as follows.

$$Recall_{lower} = \frac{TP}{Match\ Size} \quad (2.4)$$

$$Recall_{upper} = \frac{Match\ Size - FN}{Match\ Size} \quad (2.5)$$

The match size denotes the number of pairs that are classified as match.

2.5.2 Manual Review

In this experiment, a hybrid record linkage framework will employ assume the use of a manual review process to improve the performance of the automatic record linkage results by capture match pairs in the uncertain region. The purpose of the hybrid record linkage process is to retrieve optimal linkage quality while reducing the amount of manual review required. To reach the optimal record linkage quality, the automated algorithms will solve majority of the linkages that have a high probability of being classified as a match or non-match but will also allow humans to review ambiguous pairs for final verification to enhance linkage quality.

Manual review is required in a hybrid record linkage system when the automated methods are uncertain if a pair is matched or unmatched. The pairs that require manual review are determined by two thresholds, T1 and T2 (Ramezani, 2021). The thresholds, T1 and T2, were selected such that the predicted pairs outside of these thresholds were perfect on training data (PPV = NPV = 1) in the original setting (Ramezani, 2021). These two thresholds, T1 and T2, are

defined such that a probability of a predicted pair above T1 (match) or below T2 (unmatched) meet a performance criterion and do not require manual review. If the probability of a pair lies in between T1 and T2, it will be identified as "uncertain" and sent for manual review. For the purposes of the experiment, we derive a lower bound by assuming no manual review is done and an upper bound by assuming that the manual review process will correctly identify all matches and nonmatches in the uncertain area.

$$\textit{Positive Predicted Value (PPV)} = \frac{number\ of\ true\ positives}{number\ of\ positive\ calls} = \frac{TP}{TP+FP} \qquad (2.6)$$

$$\textit{Negative Predicted Value (NPV)} = \frac{number\ of\ true\ negatives}{number\ of\ negative\ calls} = \frac{TN}{TN+FN} \qquad (2.7)$$

3. RESULTS

This experiment studies how the different ML models trained on one dataset (EHR hospital data) performs on a different dataset (Voter Registry Data).

Table 3.1: Results measuring each model's performance across different heterogeneity rates

model	error rate	data size	match size	TP	FP	FN	TN	Precision	recall overall	F1	recall min	review	review %	match_size-TP	match_size-(TP-FN)	match_size-(TP-FN+TP)	recall max
rf	0	10000	1200	218	2	9	5638	0.991	0.960	0.975	0.182	4133	41.330	982	973	1191	0.993
rf	5	10042	1205	207	2	12	5691	0.990	0.945	0.967	0.172	4130	41.127	998	986	1193	0.990
rf	10	10086	1210	201	1	16	5722	0.995	0.926	0.959	0.166	4146	41.106	1009	993	1194	0.987
rf	15	10131	1213	197	2	20	5771	0.990	0.908	0.947	0.162	4141	40.875	1016	996	1193	0.984
rf	20	10181	1213	182	2	28	5867	0.989	0.867	0.924	0.150	4102	40.291	1031	1003	1185	0.977
rf	25	10232	1230	163	2	39	5908	0.988	0.807	0.888	0.133	4120	40.266	1067	1028	1191	0.968
rf	30	10290	1240	166	1	34	5984	0.994	0.830	0.905	0.134	4105	39.893	1074	1040	1206	0.973
rf	35	10344	1240	147	1	45	6054	0.993	0.766	0.865	0.119	4097	39.608	1093	1048	1195	0.964
rf	40	10412	1247	140	2	51	6093	0.986	0.733	0.841	0.112	4126	39.627	1107	1056	1196	0.959
rf	45	10486	1257	142	2	41	6246	0.986	0.776	0.869	0.113	4055	38.671	1115	1074	1216	0.967
rf	50	10578	1277	124	1	63	6300	0.992	0.663	0.795	0.097	4090	38.665	1153	1090	1214	0.951
rf	55	10677	1290	132	1	64	6398	0.992	0.673	0.802	0.102	4082	38.232	1158	1094	1226	0.950
rf	60	10807	1302	108	0	80	6562	1.000	0.574	0.730	0.083	4057	37.540	1194	1114	1222	0.939
svm_linear	0	10000	1200	702	1	19	7894	0.999	0.974	0.986	0.585	1384	13.840	498	479	1181	0.984
svm_linear	5	10042	1205	676	1	27	7952	0.999	0.962	0.980	0.561	1386	13.802	529	502	1178	0.978
svm_linear	10	10086	1210	654	0	40	8023	1.000	0.942	0.970	0.540	1369	13.573	556	516	1170	0.967
svm_linear	15	10131	1213	636	1	51	8082	0.998	0.926	0.961	0.524	1361	13.434	577	526	1162	0.958
svm_linear	20	10181	1213	595	1	69	8171	0.998	0.896	0.944	0.491	1345	13.211	618	549	1144	0.943
svm_linear	25	10232	1230	565	1	84	8229	0.998	0.871	0.930	0.459	1353	13.223	665	581	1146	0.932
svm_linear	30	10290	1240	541	0	102	8292	1.000	0.841	0.914	0.436	1355	13.168	699	597	1138	0.918
svm_linear	35	10344	1240	514	1	103	8374	0.998	0.833	0.908	0.415	1352	13.070	726	623	1137	0.917
svm_linear	40	10412	1247	503	2	133	8462	0.996	0.791	0.882	0.403	1312	12.601	744	611	1114	0.893
svm_linear	45	10486	1257	469	1	142	8590	0.998	0.768	0.868	0.373	1284	12.245	788	646	1115	0.887
svm_linear	50	10578	1277	419	1	166	8653	0.998	0.716	0.834	0.328	1339	12.658	858	692	1111	0.870
svm_linear	55	10677	1290	416	1	198	8793	0.998	0.678	0.807	0.322	1269	11.885	874	676	1092	0.847
svm_linear	60	10807	1302	379	0	250	8962	1.000	0.603	0.752	0.291	1216	11.252	923	673	1052	0.808
svm_radial	0	10000	1200	960	10	27	7209	0.990	0.973	0.981	0.800	1794	17.940	240	213	1173	0.978
svm_radial	5	10042	1205	933	10	37	7284	0.989	0.962	0.975	0.774	1778	17.706	272	235	1168	0.969
svm_radial	10	10086	1210	908	8	47	7335	0.991	0.951	0.971	0.750	1788	17.728	302	255	1163	0.961
svm_radial	15	10131	1213	897	9	52	7397	0.990	0.945	0.967	0.739	1776	17.530	316	264	1161	0.957
svm_radial	20	10181	1213	842	9	74	7486	0.989	0.919	0.953	0.694	1770	17.385	371	297	1139	0.939
svm_radial	25	10232	1230	816	9	88	7553	0.989	0.903	0.944	0.663	1766	17.260	414	326	1142	0.928
svm_radial	30	10290	1240	809	7	98	7615	0.991	0.892	0.939	0.652	1761	17.114	431	333	1142	0.921
svm_radial	35	10344	1240	778	7	107	7715	0.991	0.879	0.932	0.627	1737	16.792	462	355	1133	0.914
svm_radial	40	10412	1247	731	8	128	7801	0.989	0.851	0.915	0.586	1744	16.750	516	388	1119	0.897
svm_radial	45	10486	1257	720	7	120	7939	0.990	0.857	0.919	0.573	1700	16.212	537	417	1137	0.905
svm_radial	50	10578	1277	697	8	162	8042	0.989	0.811	0.891	0.546	1669	15.778	580	418	1115	0.873
svm_radial	55	10677	1290	660	8	176	8179	0.988	0.789	0.878	0.512	1654	15.491	630	454	1114	0.864
svm_radial	60	10807	1302	625	4	215	8354	0.994	0.744	0.851	0.480	1609	14.888	677	462	1087	0.835

In **Table 3.1**, the data shows the results of the model's performance on the Voter Registry dataset. The highlighted columns in the figure denote the four main evaluation methods that were calculated from the retrieved data.

3.1 The F1 Score of Different Machine Learning Models on a New Setting across Increasing Heterogeneity Rates

As seen in **Figure 3.1**, the radial and linear SVM models perform best in the new setting compared to the Random Forest model. The $F1_{auto}$ is reasonable for all three models remaining above 0.7 for all three models but the $F1_{auto}$ decreases for the three models across increasing heterogeneity rates with no sudden drops even with more than half the records corrupted (i.e., 60% heterogeneity). The random forest model has inconsistent F1 performance at different heterogeneity rates. However, the tradeoff between the random forest and linear SVM models remain low throughout all heterogeneity rates. Moreover, the radial SVM model has the best performance with the slowest rate of decrease compared to the other models with the $F1_{auto}$ only decreasing by 0.13 from 0% to 60% heterogeneity rate.

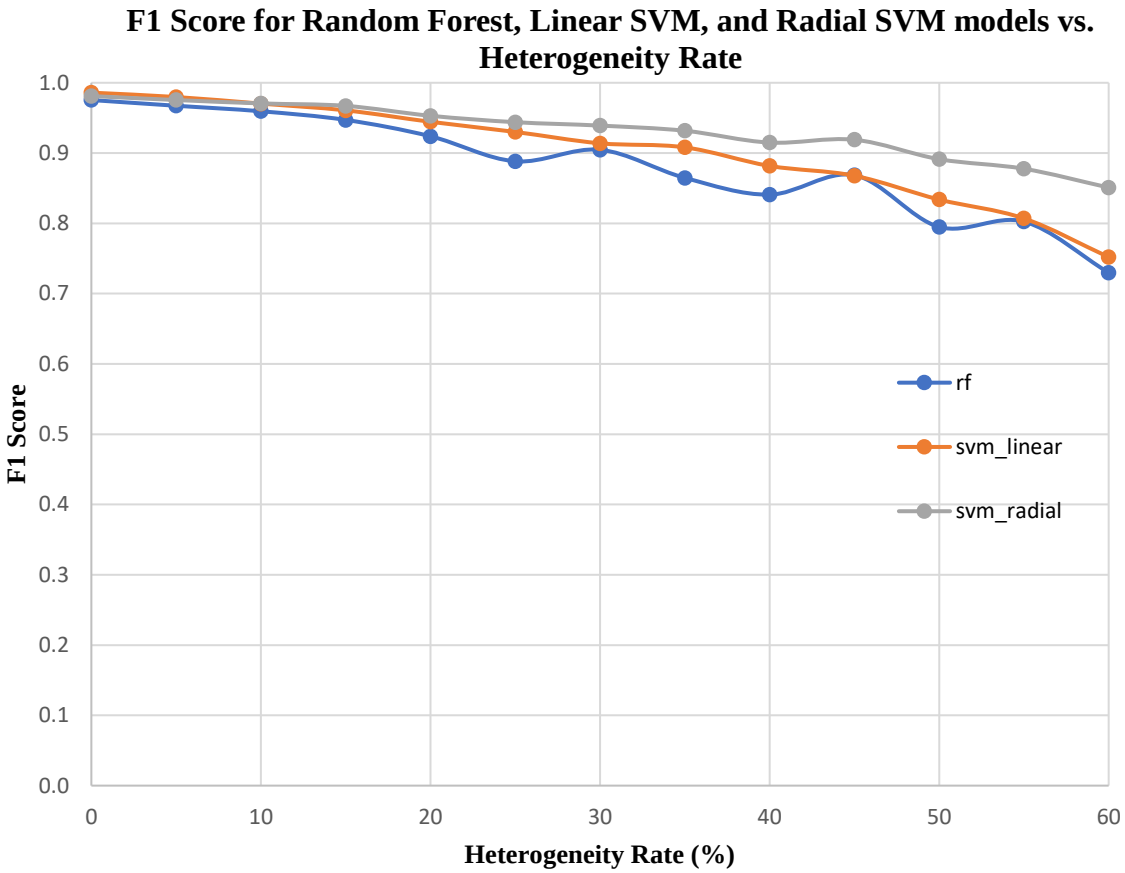

Figure 3.1: F1 Score for Random Forest, Linear SVM, and Radial SVM models from error rates 0% to 60% for automatic Record Linkage data

3.2 The Lower/Upper Bound Recall of Different Machine Models in a New Setting at Different Heterogeneity Rates

According to **Figure 3.2**, the graph displays the lower and upper bound recall for each model. Overall, the $recall_{lower}$ and the $recall_{upper}$ performance of each model decreases as the heterogeneity rate increases. However, the radial SVM model has a significantly higher $recall_{lower}$ (80% - 48%) than the linear SVM (59% - 29%) and random forest model (18%-8%) across all heterogeneity rates (0% - 60%). Although the radial SVM model may have a higher $recall_{lower}$, the random forest model had the highest $recall_{upper}$ than the other models at all

heterogeneity rates. This shows that the performance of the random forest model is the most effective when manual review is done and the performance of the radial SVM is the most effective when no manual review is done on the pairs.

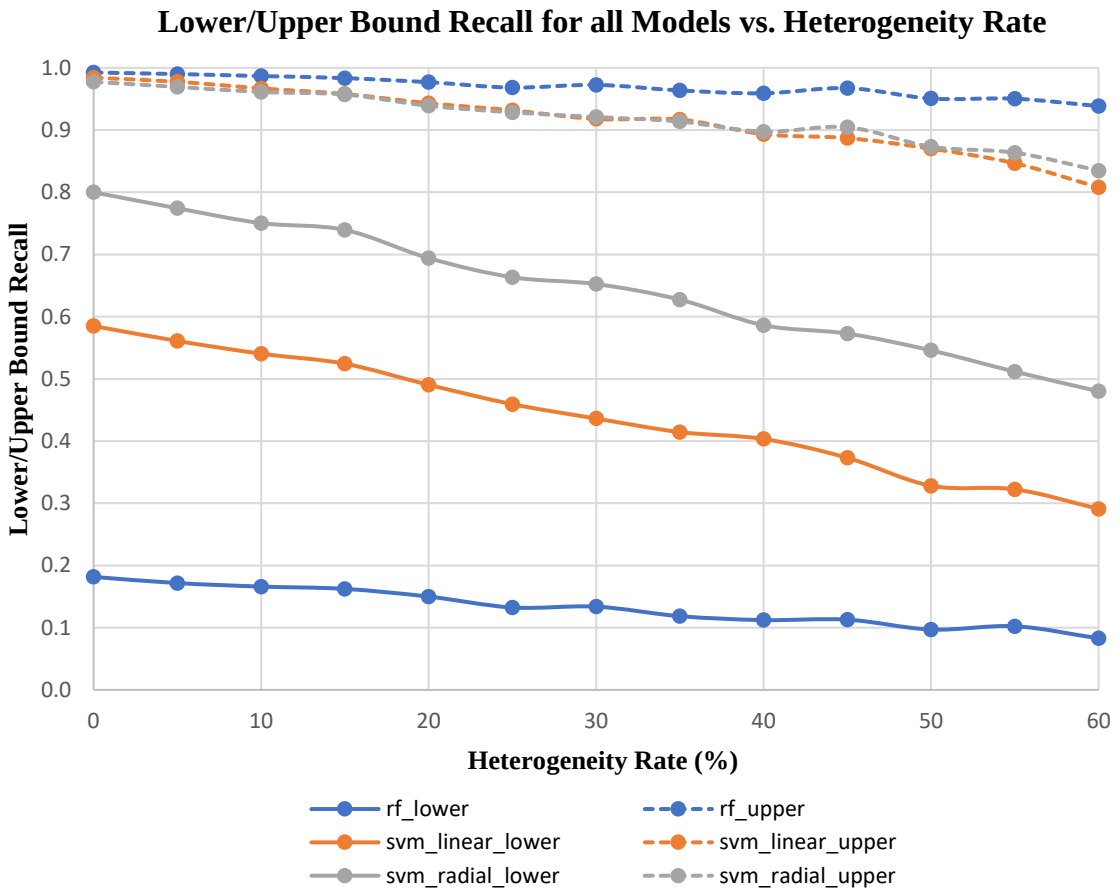

Figure 3.2: Lower/Upper Bound Recall (%) for Random Forest, SVM Linear, and SVM Radial models from error rates 0% to 60% for all data

3.3 Effect of Heterogeneity Rate on Manual Review Size on Different Machine Learning Models

As seen in **Figure 3.3**, the increase in heterogeneity rate had a minimal effect on the percentage of manual review required. The linear SVM and the radial SVM models are not far from each other in terms of manual review across all heterogeneity rates. However, the linear SVM model outperforms the random forest requiring a mean of 12.9% of manual review while the random forest model required the most amount of manual review needing around an average of 39.8% of manual review.

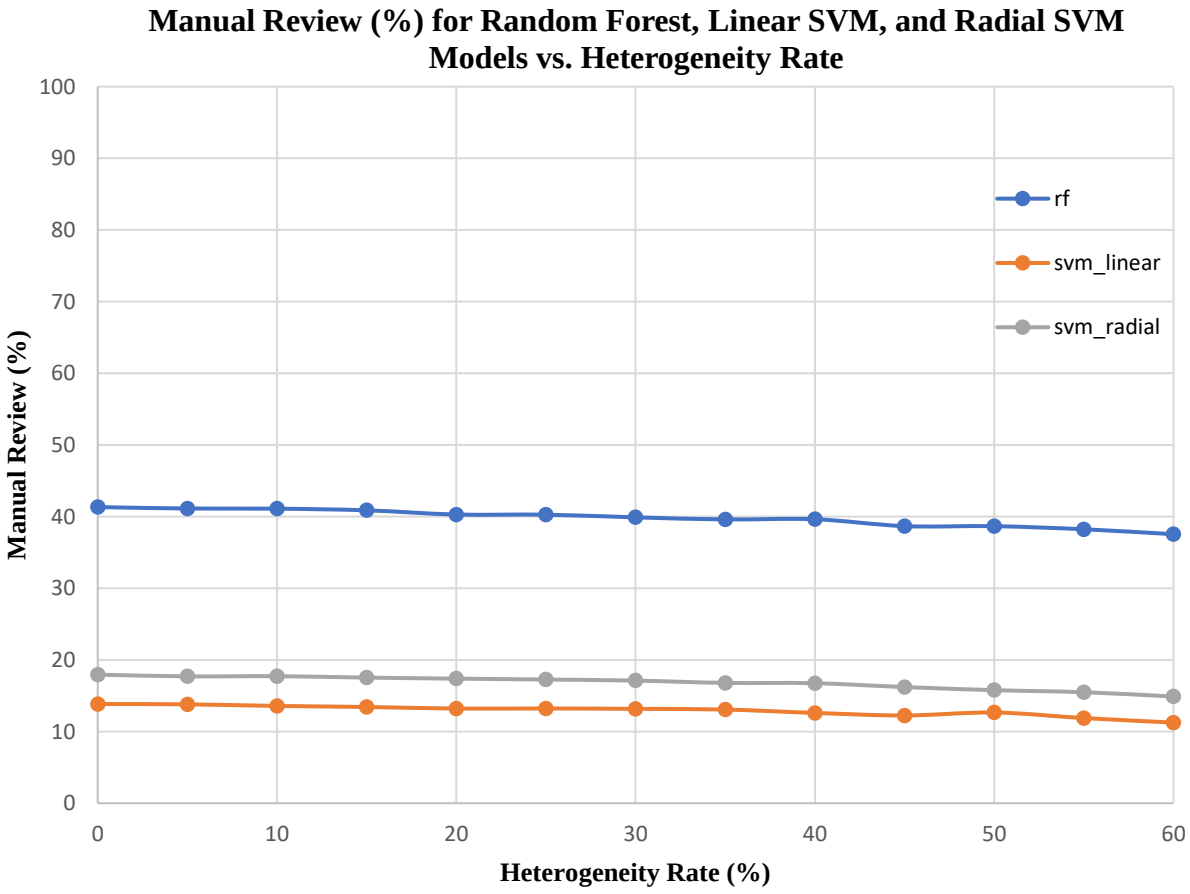

Figure 3.3: Manual Review (%) for Random Forest, SVM Linear, SVM Radial, and Neural network models from error rates 0% to 60% for automatic Record Linkage data

4. CONCLUSION

4.1 Limitations

There are several limitations in this study. First, we need to run more experiments to understand which characteristics of the SVM models make it more effective in a hybrid record linkage system. Second, each model was only trained on one type of dataset (EHR Data) and was tested and evaluated on one other type of dataset (Voter Registry Data). Transferability of the models will depend on the similarity of the training data and testing data. Thus, more experiments with different types of data are needed to assess whether our findings are generalizable to more data types.

4.2 Discussion

As the amount of data grows largely, data integration becomes a critical issue to solve because it is important to maintain the quality of the linked results. Automatic record linkage has progressed over the past few years but does not still have the same reliability of manual record linkage. However, the manual record linkage process can be time consuming. Therefore, in this research, an open-source hybrid record linkage framework was presented that combines the automated and manual process to achieve high quality linkage results. This research studies the performance of three different ML algorithms (Random Forest, Linear SVM, and Radial SVM) on different settings in a hybrid record linkage system at various heterogeneity rates to better inform which ML models should be used in different circumstances.

In this study, we first evaluated the three models using the F1 measure for the automated part, $F1_{auto}$. As the heterogeneity rate increases, the $F1_{auto}$ performance of every model degrades somewhat with the radial SVM having slightly better $F1_{auto}$ score among the three models across

all heterogeneity rates. In terms of recall, again the radial SVM model had the best performance with substantially better $recall_{lower}$ than the other two models. The random forest and linear SVM had a mean $recall_{lower}$ percent of 13% and 44% respectively and the radial SVM had a mean $recall_{lower}$ percent of 65% across all heterogeneity rates. The radial SVM outperforms the two models by 36% on average which shows how accurate the model is able to identify and linkages when no manual review is done. When manual review is done, all three models have the potential to recover most linkages and improve $recall_{upper}$ up to 96.9% (random forest), 91.9% (radial SVM), and 91.5% (linear SVM) on average. Although the random forest model had the best $recall_{upper}$ performance, it would require a lot of manual review with an average percentage of 39.8%. That is 2.3 and 3 times more review for radial and linear SVM respectively. In comparison, radial SVM could achieve comparable $recall_{upper}$ for only 16.8% manual review. Of note, the increasing heterogeneity rate had little impact on the manual review percentage across all three models indicating that the ML models were effective in resolving most of the heterogeneities introduced. In sum, with the $F1_{auto}$ ranging from 0.94 to 0.85 and $recall_{lower}$ ranging from 0.65 to 0.48 from 30% to 60% heterogeneity rate for the radial SVM model, we can conclude that the radial SVM model does transfer to the new setting fairly well.

REFERENCES

[1] Ilangovan, G. (2019). Benchmarking the Effectiveness and Efficiency of Machine Learning Algorithms for Record Linkage. Master's thesis, Texas A&M University.

[2] Kaur, P. (2020). A Comparison Of Machine Learning Classifier For Use On Historical Record Linkage. Master’s thesis, The University of Guelph.

[3] NCSBE. (n.d.). Retrieved August 29, 2021, from https://www.ncsbe.gov/results-data/voter-registration-data

[4] Ramezani F., M. (2021). Comparison of Machine Learning Algorithms in a Human-Computer Hybrid Record Linkage System. Master's thesis, Texas A&M University.