Exploring the Potential of Probabilistic Record Linkage in Healthcare

A Study on Matching National Provider Identifier Records with Social Network Profiles

Florian Jürgen Pullem

Dissertation written under the supervision of Professor Nicolò Bertani

Dissertation submitted in partial fulfilment of requirements for the MSc in Business Analytics, at the Universidade Católica Portuguesa, 03.01.2024.

Abstract

Title: Exploring the Potential of Probabilistic Record Linkage in Healthcare:
A Study on Matching National Provider Identifier Records with Social Network Profiles
Author: Florian Jürgen Pullem

In the digital era, a wealth of heterogeneous data is collected globally about various entities such as individuals, professionals, or companies. Extracting value from this data requires linking individual data points that describe the same entity. However, the diversity of sources and the absence of a unique identifier complicate this process.

This study addresses this challenge by exploring the potential of probabilistic record linkage techniques to associate entries in the National Provider Identifier (NPI) database with physician's social network profiles. The research was conducted in collaboration with Alpha Sophia, a startup aiming to build a leading commercial intelligence platform for the US healthcare market.

The thesis proposes an innovative strategy for generating labeled data, which comprises a combination of deterministic record linkage and noise injection. This strategy facilitates the implementation of various supervised learning models, such as random forest, alongside the benchmark, the Fellegi-Sunter model.

The primary finding is the superior performance of supervised models over the benchmark, demonstrating the advantage of the innovative approach. Over 142 thousand new matches were identified while maintaining a minimal false positive rate. This equates to an approximate 64% increase in the total number of linked data records compared to the number of matches discovered through traditional methods. Moreover, cost savings exceeding 68 thousand euros were realized.

The methodologies and model presented can be tailored to address other linkage challenges that Alpha Sophia and other companies encounter. It is recommended to employ the outlined techniques in diverse contexts with varying datasets in the future.

Keywords: Probabilistic Record Linkage, Noise Injection, National Provider Identifier (NPI), Fellegi-Sunter Model, Logistic Regression, Random Forest

Abstract Portuguese

Título: Explorando o Potencial da Ligação de Registros Probabilísticos na Saúde:
Um Estudo sobre a Correspondência de Registros de Identificadores de Prestadores Nacionais com Perfis de Redes Sociais
Autor: Florian Jürgen Pullem

Na era digital, uma grande quantidade de dados heterogéneos é coletada globalmente sobre várias entidades, como indivíduos, profissionais ou empresas. Extrair valor desses dados requer a ligação de pontos de dados que descrevem a mesma entidade, um processo complicado pela diversidade de fontes e ausência de um identificador único.

Este estudo aborda este desafio ao explorar o potencial das técnicas de ligação probabilística de registos para associar entradas na base de dados do NPI aos perfis das redes sociais dos médicos. A investigação foi realizada em colaboração com a Alpha Sophia, uma startup visando criar uma plataforma de inteligência comercial para o mercado de saúde nos EUA.

A tese propõe uma estratégia inovadora para gerar dados rotulados, combinando ligação determinística de registos e injeção de ruído. Esta estratégia facilita a implementação de vários modelos de aprendizagem supervisionada, como a floresta aleatória, ao lado do modelo Fellegi-Sunter.

A descoberta principal é o desempenho superior dos modelos supervisionados em relação ao modelo de referência, demonstrando a vantagem da abordagem inovadora. Foram identificadas mais de 142 mil novas correspondências, mantendo uma taxa mínima de falsos positivos. Isto equivale a um aumento de cerca de 64% no número total de registos de dados ligados, resultando em poupanças de custos superiores a 68 mil euros.

As metodologias e o modelo apresentados podem ser adaptados para responder a outros desafios de ligação que a Alpha Sophia e outras empresas possam enfrentar. Recomenda-se a aplicação destas técnicas em contextos diversos com conjuntos de dados variados no futuro.

Palavras-Chave: Probabilistic Record Linkage, Noise Injection, National Provider Identifier (NPI), Fellegi-Sunter Model, Logistic Regression, Random Forest

Acknowledgments

First and foremost, I would like to express my deepest gratitude to my family and friends. Their support and encouragement throughout my academic journey have been invaluable. Their faith in my abilities and their understanding during challenging times have played a significant role in my accomplishments.

I am profoundly grateful to my professor, Nicolò Bertani, for his guidance and support throughout the course of my master's program and the supervision of this thesis. His expertise and dedication have been instrumental in the successful completion of this research.

I would also like to extend my appreciation to Alpha Sophia for providing an innovative challenge, invaluable data, and their support. The opportunity to work on a real-world problem has enriched my learning experience and has been incredibly rewarding.

My sincere thanks go to the developers of Python and the creators of the packages that were instrumental in this research. These include NumPy, Matplotlib, Jellyfish, Scikit-learn, RecordLinkage, Pandas, Seaborn, Random, PySpark, and Splink. Their contributions to the open-source community have made this work possible.

Finally, I would like to acknowledge OpenAI for the supply of ChatGPT, which was used to enhance the clarity and formulation of this thesis. It is important to note that while ChatGPT was utilized to optimize the presentation of the content, the thoughts, ideas, and knowledge presented in this thesis are entirely my own, except where explicitly cited or referenced. If demanded, the prompts used for this purpose can be provided.

Contents

List of Figures

List of Tables

List of Formulas

List of Abbreviations

Abbreviation	Definition
AWS	Amazon Web Services
CMS	Centers for Medicare and Medicaid Services
EM	Expected Maximization
NFD	Normalization Form Canonical Decomposition
NPI	National Provider Identifier
PC	Pair Completeness
PQ	Pair Quality
PPRL	Privacy-Preserving Record Linkage
PR	Precision-Recall
RF	Reduction Factor
ROC	Receiver Operating Characteristics
RR	Reduction Ratio
SN	Social Network
SVM	Support Vector Machines
USA	United States of America

1 Introduction

Linking data that describe the same entity but vary in structure and quality is a pivotal task in the digital age. In the burgeoning research field of probabilistic record linkage, predictive models are applied to overcome this obstacle. As there is no one-size-fits-all solution, the primary challenge consists of the selection of the appropriate method. In this research, various methods are explored to answer the research question faced by the startup Alpha Sophia:

"How can probabilistic record linkage methods be applied to effectively and accurately match National Provider Identifier (NPI) records with social network profiles of healthcare providers?"

1.1 Business Case

Alpha Sophia is on a mission to build the leading commercial intelligence platform for the United States healthcare market. They aim to empower innovators in the medical device, pharmaceutical, and life science sectors to identify and connect with relevant healthcare providers. This includes physicians and organizations essential for bringing their clinical trial, medical device, or drug to the market and ultimately to the patient. The company's business model consists of linking and aggregating structured and unstructured datasets from various sources, including governmental agencies and public websites. A significant part stems from the integration of social media profiles and related content.

However, Alpha Sophia faces a common challenge: Traditional methods such as deterministic linking based on specific attributes (e.g., names and addresses) often prove insufficient. They frequently leave a significant portion of a dataset unmatched and therefore underutilized in terms of commercial value. This issue primarily arises from data across various sources that may be outdated, inaccurately maintained, fuzzy (such as addresses, which come in various formats like "15 West Kings Street, Brooklyn" vs. "15 W Kings St, New York"), or contain typos.

In a recent scenario, Alpha Sophia collected a substantial amount of fuzzy data on physicians from social network websites. After using deterministic linking techniques, almost 304 thousand out of 534 thousand profiles (approximately 57%) remained unmatched. The transition to machine learning models might improve the number of matches. To estimate the commercial impact, two dimensions come to mind:

- Loss of customer conversion for Alpha Sophia due to insufficient social media coverage
- Costs of labor to bridge the matching gap with manual matching

Alpha Sophia's target market includes over 6,500 potential medical device companies (AdvaMed, 2023). Providing information about multiple contact channels for each physician would facilitate effective omnichannel marketing for Alpha Sophia's customers, adding significant value (Harrison et al., 2021). Internal calculations suggest that nearly 8% of potential customers are lost due to inadequate social media coverage. Given an annual base subscription fee of 3,600 euros per customer, this translates to a potential yearly revenue loss of 1.872 million euros across the entire market.

Linking the remaining profiles, as committed in Alpha Sophia's value proposition, would require substantial time and resources if done manually. Preliminary tests indicate that the manual linkage of a profile takes about two minutes on average. The primary challenge is that the matching status of the remaining profiles is often unclear, and in some cases, a suitable link may not exist. This results in an investment of approximately 3,333 hours per 100 thousand new links. In Germany, an hour of minimum wage labor costs the employer at least 14.44 euros (Handelsblatt, 2023). Even at this conservative wage estimate, the cost per 100 thousand links would be approximately 48,129 euros, excluding additional employer expenses such as vacation bonuses and sick leave. Furthermore, a full-time employee would spend over 19 months on this task alone.

Considering these figures, it becomes evident that a different approach is imperative. Probabilistic record linkage presents an alternative solution, potentially leading to significant cost and time savings for Alpha Sophia.

1.2 Summary of Work and Findings

This thesis addresses Alpha Sophia's business problem by exploring and comparing various probabilistic record linkage models. The initial step involves a review of relevant literature in the field of record linkage, establishing the foundation for the methodology of this thesis.

After a thorough examination and preparation of the available datasets, including standardization and blocking, the widely accepted Fellegi-Sunter model is implemented. An innovative approach is then introduced to convert the unsupervised problem into a supervised one using deterministic record linkage techniques and noise injection. The presence of labeled data enables the application of supervised classification models, such as logistic regression and random forest.

Finally, a variety of performance metrics like precision or recall are employed to facilitate an objective comparison. These metrics, evaluated at different classification thresholds, consistently demonstrate superior performance of the random forest.

In this research, nearly 143 thousand new matches were identified, while maintaining a minimal false positive rate. When compared with the costs presented earlier, this translates into substantial savings of 68,753 euros for Alpha Sophia. Consequently, a satisfactory solution has been identified, providing a comprehensive answer to the research question.

2 Literature Review

Record Linkage is commonly defined as the process of identifying records that pertain to the same entity and are dispersed across datasets (Christen and Winkler, 2017). It is utilized in various fields such as public health (Fienberg and Manrique-Vallier, 2009; Fleming et al., 2012), official statistics (Taylor et al., 2023), or fraud detection (Vatsalan, Sehili, et al., 2017) and plays a crucial role in many machine learning and data science projects. Typically, record linkage forms the foundation of subsequent analysis projects and enables more effective data utilization.

Despite the importance, record linkage presents several challenges such as the need for accurate and efficient methods, as well as concerns related to privacy. This literature review provides an overview of the current research in record linkage, encompassing the historical context, prevalent methods, and potential future directions.

2.1 Development of Record Linkage

The concept of record linkage, as understood today, traces back to 1946 when Halbert L. Dunn described it as the "process of assembling the pages of each person´s book of life into a volume" (Dunn, 1946, p. 1412). The scattered data, if correctly linked, can yield significant value for science and the economy. Dunn highlighted the primary challenge behind record linkage: the fragmentation of an individual's data across various locations. While he mainly referred to physical locations, this statement extends to digital data traces in times of digitalization.

Dunn's initial process, now known as deterministic or rules-based record linkage, involves linking records based on matching identifiers in both datasets. It is particularly suitable for high-quality data.(Roos and Wajda, 1991). However, as deterministic record linkage was commonly used to link major life events such as birth, marriage, and death, additional challenges emerged. Kilss and Alvey (1985) identified several issues in data quality making a deterministic approach less effective. These include spelling errors, varying data structures, and insufficient identifying information for an accurate linkage.

To address data discrepancies, they introduced an alternative approach, known as probabilistic record linkage. This method assigns a match probability to a pair of records based on the agreement or calculated distance between multiple attributes. Probabilistic record linkage is used in the subsequent work to solve Alpha Sophia's business problem.

2.2 Probabilistic Record Linkage

The procedure of probabilistic record linkages involves several core elements presented in Figure 2.1 (Kevin et al., 2019).

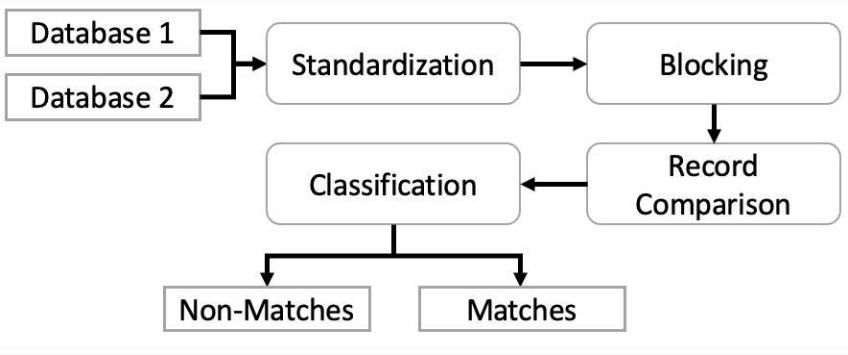

Figure 2.1: Record Linkage Process, Adapted from Kevin et al. (2019, p. 80).

The initial steps include cleaning and standardizing the obtained data to remove inconsistencies between attributes that should match (ibid.). Given that data records originate from diverse sources, the information within these records can be structured and formatted differently. For instance, one dataset might contain addresses inserted manually without any formatting rules (e.g., Main str. 83, Sunshine Blv 2), while another dataset adheres to fixed formatting rules (e.g., Main Street 83; Sunshine Boulevard 2). The goal of standardization is to make datasets as comparable as possible, thereby facilitating the identification of similarities between individual items. For an accurate record linkage, standardization can be more crucial than improving the model or parameters itself (Winkler, 2014).

Record linkage is typically performed on a one-to-one basis, where each entry from one dataset is compared with each entry of another dataset, assigning a match probability to each pair. With a corresponding table size, this can lead to a large number of comparisons (Kevin et al., 2019). For instance, linking two datasets with a million entries each would require a trillion comparisons, a challenging task even with today's computing power. To manage this, a process known as blocking is employed. Blocking reduces the number of comparisons by only comparing data entries that are equal on a predefined basis. The first blocking procedures were described by Newcombe (1967). In this traditional blocking process, only entries that match in certain characteristics such as gender or city (and therefore share the same blocking key) are compared with each other.

Blocking in record linkage has evolved into a research field of its own, with various deterministic (e.g., traditional blocking, hashing) and non-deterministic (e.g., clustering) methods in use (Steorts et al., 2014). The result of the blocking process is a table (or comparison vector), with each row containing a pair of records that remain to be compared. While the blocking methods

differ in complexity and computational requirements, their performance is typically measured on three levels (Nowak et al., 2021):

1. **The Reduction Ratio (RR)** describes the relative reduction of the amount of necessary comparisons and is defined as:

$$RR = 1 - \frac{s}{nM + nU} \tag{2.1}$$

, where nM and nU describe the number of total matches and non-matches without blocking and s the size of the comparison vector after blocking (Elfeky et al., 2002).

2. **The Pair Completeness (PC)** is the ratio of matches sM that still exists in the comparison vector after the blocking process and is calculated as follows:

$$PC = \frac{sM}{nM} \tag{2.2}$$

3. **The Pair Quality (PQ)** describes the proportion of true matches in the resulting comparison vector and is defined as:

$$PQ = \frac{sM}{sM + sN} \tag{2.3}$$

Once a dataframe including the potential matches is created, the next step is to make the opposing information comparable. Numerous methods are available, which vary in their effectiveness when comparing different types of attributes. While it might be sensible to check for exact matches of the categorical variable "city", the "zip code" is often compared using the Hamming Distance (ibid.). In the context of the Hamming Distance, the count of mismatches matters. The postcodes 50935 and 51933 would exhibit two mismatches.

Additionally, multiple metrics are available for the comparison of text attributes with variable lengths. The Edit or Levenshtein Distance describes the minimum steps required to convert one string into another, considering insertions, deletions, or replacements (Elfeky et al., 2002; Navarro, 2001). For instance, the Edit Distance between “Parker Street 13” and “Park Street 24” is 4. Moreover, the Jaro-Winkler Distance has gained considerable popularity (Elfeky et al., 2002; Jaro, 1976). This measure gives more weight to the prefix of the strings, as studies have shown that fewer errors typically occur at the beginning of strings (Christen, 2006; Pollock and Zamora, 1984). The metrics used in this study are detailed in Section 5.2.

The classification process itself can be accomplished using various machine-learning models. When labeled data is available, a range of supervised models such as logistic regression, decision trees, or support vector machines (SVM) are typically employed (Asher et al., 2020). The

selection of the appropriate model depends on several factors such as the structure, quality, and quantity of training data.

However, instances where labeled data is available are rare and the procurement of such data is often costly in terms of time and effort (Cao et al., 2011). As a consequence, unsupervised models are frequently used for record classification. In these cases, various clustering techniques like the Expected Maximization (EM) algorithm and the Fellegi Sunter Model are employed. The Fellegi-Sunter model, introduced by Fellegi and Sunter (1969), is a widely used tool for record linkage. It calculates agreement and disagreement scores for each field between two records. These scores are weighted and summed up, and the resulting total score is used to classify the pair of records as a match, non-match, or possible match based on given thresholds. A more detailed explanation of this model can be found in Section 5.1.1.

The evaluation of the linkage procedure depends on the model type. For supervised models, standard classification metrics like precision, recall, and the F-score are typically utilized (Chipperfield et al., 2018; Hand and Christen, 2018). These metrics measure the accuracy and completeness of the classification performed by the model. Evaluating unsupervised models presents a more complex challenge. Strategies may include comparing matched and unmatched records to identify bias or assessing result sensitivity to minor procedure adjustments (Steorts et al., 2014).

2.3 Current Trends and Future Directions

With the advent of increased computational power and data accumulation, research in record linkage is progressing rapidly (Asher et al., 2020). Current developments focus on leveraging advanced machine learning methods to enhance the effectiveness and evaluation capabilities of the linkage process.

Significant research is being conducted on Bayesian models due to their ability to efficiently handle large volumes of data and complex relationships (Taylor et al., 2023). These models incorporate prior knowledge about parameters based on previous data or expert knowledge into the model. This knowledge is updated as new data is observed, providing a powerful and flexible approach to handle uncertainty and complexity (Gelman et al., 2013).

Furthermore, making record linkage techniques more accessible for organizations is a key area of focus. To facilitate the linkage process, a wide array of applications, tools, and packages are being developed. A notable example is the Record Linkage Toolkit (Bruin, 2019). This toolkit allows users to construct Python programs and provides access to open-source packages like

"fastLink" (Enamorado et al., 2017). These advancements allow more effective and user-friendly record linkage, paving the way for future innovations in this field.

As in many other research areas dealing with sensible data, Privacy-Preserving Record Linkage (PPRL) is a burgeoning trend in record linkage research (Khan et al., 2022; Vatsalan, Karapiperis, et al., 2022). Given that the entities are often individuals, data protection has to be considered critically. This is particularly true in medical research where patient data, such as prescribed medications, must be handled confidentially. Various techniques have been developed to reconcile the need for data linkage with privacy protection (Gkoulalas-Divanis et al., 2021). These techniques either transform the data or facilitate privacy-preserving data exchange. Cryptographic hashing, for instance, transforms data points into a hash that represents the original data, but from which the original data cannot be feasibly derived (Adir et al., 2022).

In the context of this work, it is important to note that privacy-preserving techniques were not employed. This is because all data used is publicly accessible and represents individuals in their professional roles as physicians, not in their personal capacity. Consequently, no private addresses or sensitive information were handled.

This work endeavors to present an innovative solution to the prevalent challenge of the unavailability of labeled data. It elucidates the conditions and methods through which labeled data can be generated autonomously, thereby transforming an unsupervised problem into a supervised one. The widely recognized Fellegi-Sunter model serves as a benchmark in this study, demonstrating how alternative labeling techniques can enhance the performance of the record linkage process.

3 Data

The datasets for this study, provided by Alpha Sophia, were obtained via Snowflake. Two main tables were extracted by merging various fact and dimension tables:

- The **National Provider Identifier (NPI)** table with information about physicians practicing in the US, initially provided by the Centers for Medicare and Medicaid Services (CMS) (Medicare and Services, 2023).
- The **Social Network (SN)** table provides information about physician's professional Social Network representations.

This chapter addresses the type and quality of the datasets, evaluated through an extensive explanatory analysis.

3.1 NPI Table

The NPI table, extracted in July 2023, includes all physicians and medical organizations practicing in the USA, along with their NPI identification number and contact information. To reduce computational costs, deactivated NPIs and irrelevant entries were filtered out during the export from Snowflake. These included entries describing organizations or entities other than (future) physicians or nurses that are allowed to prescribe. Based on these filter criteria, the initial table of 7.9 million entries was reduced to about 2.3 million entries.

The NPI table encompasses a broad range of attributes with the 10-digit NPI code as a unique identifier. Of the approximately 2.3 million physicians, 1.23 million are male. A physician's name is divided into "first_name", "middle_name", and "last_name". The middle name contains about 828 thousand missing entries, while both other name fields are fully populated. The quality of the name information is generally good, but occasionally there are multiple names in one field or only the first letter of a name is provided. Additional fields for prefixes, suffixes, and credentials also contain numerous missing values.

The dataset provides address fields, categorized as mailing and practice addresses. Both are divided into multiple parts: "street_first" ideally includes the street name and house number, while "street_second" provides details like suite or flat numbers. Further address information is given in the "zip", "city", "state", "country", "phone" and "fax" fields. While mailing and practice addresses are mostly identical, discrepancies mainly occur in the street information. The mailing address tends to have more missing entries, particularly in the second street part, phone,

and fax fields. The overall quality of the address information could be enhanced, considering the significant variation in the street fields, as well as the presence of numerous entries that diverge from the expected schema.

The NPI table was enriched with "taxonomy" (specialization) information from another table. In cases where physicians were assigned to multiple taxonomies, a primary taxonomy was selected rule-based in consultation with Alpha Sophia. The taxonomy is given in different grouping levels. While the overall taxonomy contains no missing entries and 45 distinct categories, the more detailed specialization level contains approximately 1.08 million missing entries. Further investigation is required to determine the level, at which NPI and SN taxonomies align most.

3.2 SN Table

The SN table is already refined to the target entities, such as (future) physicians and nurses. It comprises approximately 780 thousand records, detailing names, contact information, and specializations.

Along with the Record-ID, which is the unique identifier, there are individual fields for "first_name", "middle_name", "last_name", and "credential". The first and second names show no missing values, while the middle name is not provided for 527 thousand out of 778 thousand entries, a higher ratio as in the NPI table. There was one exact duplicate over all columns which has been removed.

The address fields follow a similar structure as in the NPI table, with attributes for street name and number ("street_first") and additional information ("street_second"), along with information about "zip", "city", "state", "phone", and "fax". Unlike the NPI table, the SN table contained 2,509 entries (approximately 0.3%) with missing address information. Given the importance of the address in the matching process, these entries were excluded. Furthermore, there is no phone number given for around 99 thousand entries.

The taxonomy information is represented by 57 distinct categories. These specializations only partially align with the taxonomies from the NPI table and will require standardization in subsequent data preparation steps.

Additionally, there are multiple entries for some physicians that link to different URLs on the social network. Further investigations have shown that there are indeed multiple profiles for some physicians, from which only one is maintained by the individual. After intensive research, a marker for the “inactive profiles” was identified. The "url" for the profile picture consists of a

placeholder for all physicians, who have not uploaded a picture yet, thus the URL either contains the string “profile_placeholder_unregistered” or “profile_placeholder_registered”. Since Alpha Sophia is not interested in linking profiles of physicians who never used the social network, approximately 242 thousand inactive profiles were filtered out. Overall, this leads to 533,744 remaining entries that are kept for standardizing, blocking, and matching.

4 Data Preparation

As outlined in Section 2.2, the record linkage process involves several steps. Following data retrieval and examination, preparatory actions like standardization and blocking were carried out.

Additionally, to create data that can be used for model training and evaluation, thereby transforming the unsupervised problem into a supervised one, further steps were executed. These encompass deterministic record linkage followed by noise injection to align the deterministically matchable data to the unmatchable data. All performed preparatory steps are described in this chapter.

4.1 Standardization

The first step of standardization deals with the taxonomy of each physician in the NPI and SN table. Detailed information about the structure and assignment of taxonomies for the NPI data are provided by the NPI Registry (Committee, 2023). The SN taxonomy seems to be influenced by the NPI structure, but there are unique categories in both datasets. The objective of the standardization is to ensure consistent categorization across both datasets. To achieve this, an in-depth analysis of the categories that showed discrepancies was conducted.

For unique categories in the SN database (35 out of 57), the CSM Taxonomy Code Set (ibid.) was examined, and the categories were matched to the NPI taxonomy structure. Most of these categories correspond to the NPI taxonomy "Internal Medicine", as the SN categorization seems to be more detailed here. However, matching on broader categories is more practical due to the numerous missing values in the more detailed categories. Since the NPI taxonomy provides clear distinctions, the assignment could be executed with high accuracy.

Aligning unique NPI taxonomy categories with existing SN taxonomies was challenging due to the absence of comprehensive SN category documentation. Investigations identified multiple categories where the NPI Taxonomy was broader (e.g., "Plastic Surgery" vs. "Surgery"), that could be assigned without further expert knowledge. For instances where the assignment was ambiguous or unclear, the NPI categories were assigned to the "Other" taxonomy. This was particularly noticeable in the absence of a category for dentists in the SN categorization, leading to approximately 250 thousand entries from the NPI table being categorized as "Other".

To ensure comprehensive coverage of all individuals represented on the social network platform, additional taxonomies such as "Registered Nurse" and "Student" were included in the NPI table. However, these categories are underrepresented in the SN table, resulting in a significantly higher proportion of "Other" in the NPI table. In total, the "Other" taxonomy encompasses 821,404 entries in the NPI database. Following standardization, Figure 4.1 illustrates the relative distribution of the 25 resulting taxonomy categories. The distribution is similar across most categories, with exceptions for "Other" and "Anesthesiology". The increased representation in the "Anesthesiology" category can be attributed to the inclusion of registered nurses and aligns with expectations as described above.

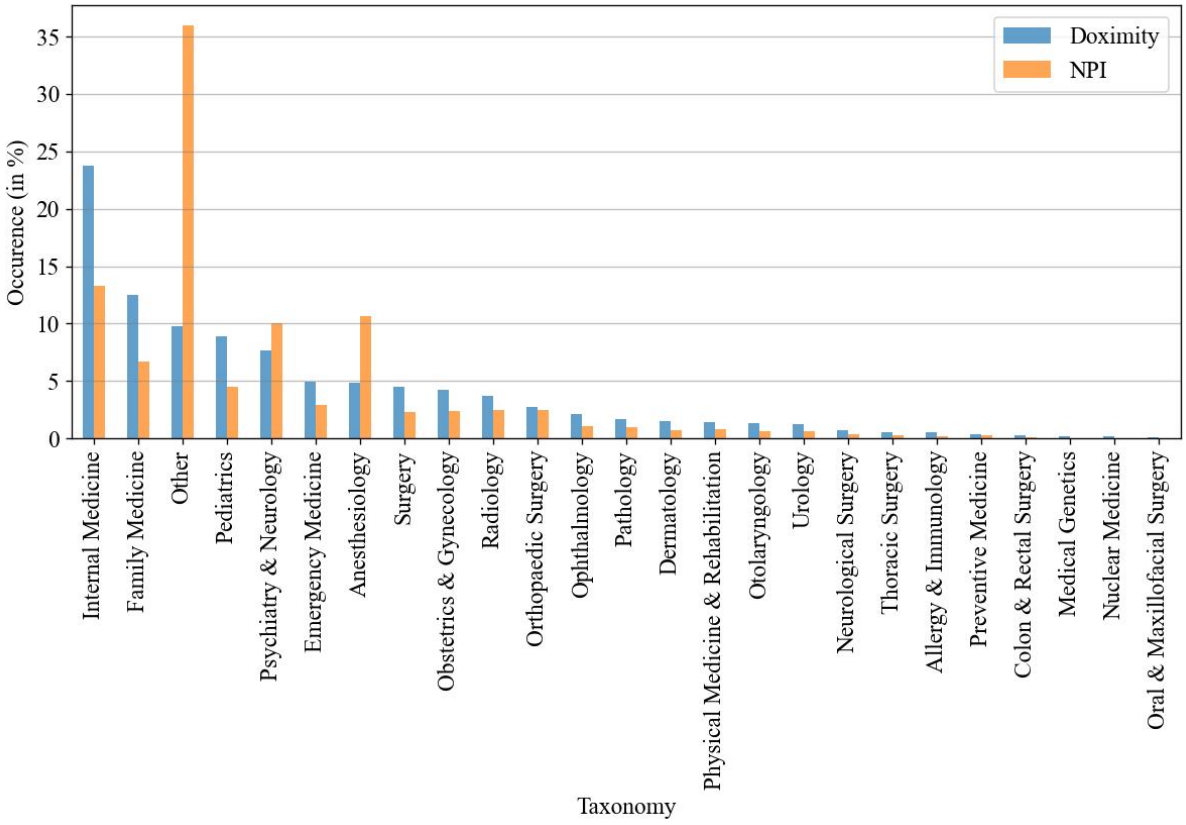

Figure 4.1: Relative Distribution of Taxonomies Across Datasets

The subsequent phase of standardization focused on the name attributes across both dataframes. All characters were converted to lowercase, addressing inconsistencies in character formats between the tables. Afterwards, the Unicode Normalization Form (NFD) was used to decompose characters with diacritical marks such as "ä" or "é" into their base character and a separate mark. Certain name entries included special characters that appeared only in one table or at different frequencies. Thus, the special character "-" was replaced with a space, and the "ł" was replaced with "l". Other special characters like dots, commas, brackets, and the decomposed diacritical marks were removed from all name fields.

Despite the presence of separate fields for suffixes, some could be still found in the name columns. These Suffixes (e.g., "Esq", "PhD", "Jr", "II") were excluded accordingly.

In conclusion, the standardization of the name attributes was relatively straightforward. After standardizing all relevant columns, a new field was created with the concatenation of all three names parts. The rationale behind this step is to ensure that the distribution of name parts into the fields does not impact the comparison of names. As a result, the placement of the middle name, whether as a second part of the first name or in the middle name field, does not affect the calculation of name similarity. Considering the name entry "John A. Doe Jr.", after standardization it would be given as "john a doe" in the "full_name" field.

Standardizing the address information was more extensive, comprising street information and city. Initially, similar to the name standardization, the entries were converted to lowercase, and special characters were removed. A detailed examination of differing street information between the two dataframes revealed that the use of abbreviations such as “plz” for “plaza” or “st” for “street” is common in both databases but not always used consistently for corresponding entries. Therefore, all entries were converted to the short form. Additionally, information that belongs to the second part of the street field, such as suite or apartment number, was removed from the first address part. The standardization was performed with a function applied to the “street_first” entry of both dataframes, resulting in a new column attribute “address”.

The standardization procedure for the city was similar to the one described above. After removing punctuations and applying substitutions, common differences were investigated. The most frequent differences were caused by different formats (e.g., "st" vs "saint", "twp" vs "township"), and local place names instead of city names (e.g., “hollywood” vs “los angeles”, “kendall” vs “miami”). All entries were synchronized to make both city columns as similar as possible.

Figure 4.2 illustrates the distribution of the 25 most occurring cities for both dataframes after standardization. The cities New York City and Washington DC are presented as "new york" and "washington". With “new york” as the most common city, the distribution is similar between both dataframes. On average, the NPI city distribution exhibits greater diversity and a higher number of special cases that were not captured during standardization, resulting in a reduced proportion of the most frequent cases.

Finally, the columns containing information about telephone numbers and zip codes were cleaned and reformatted. This involved minor adjustments, such as the removal of the country code from the phone number. Following the standardizing of the attributes pertinent to this study, the creation of the training and test datasets can now be approached.

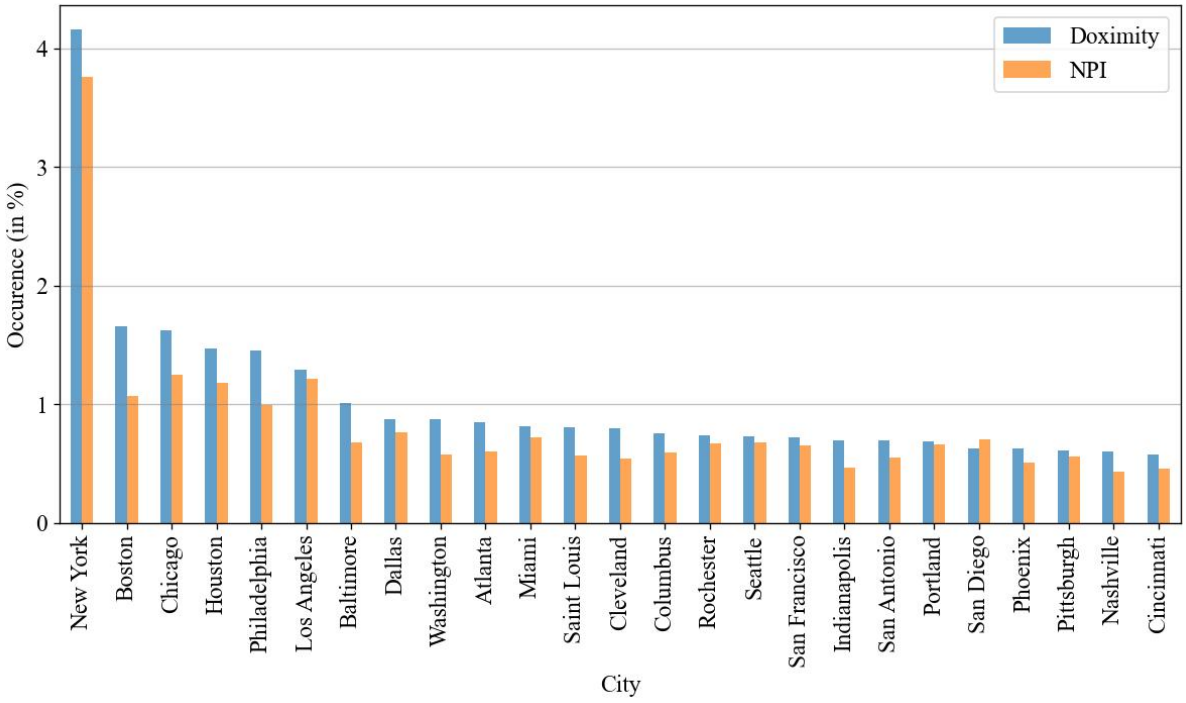

Figure 4.2: Relative Distribution of Cities Across Datasets

4.2 Data Labeling

Deterministic record linkage is utilizable for creating a training dataset, given unique identifiers for some matches. For the provided data, it is reasonable to assume that a simultaneous match of first name, last name, and phone number indicates the same individual. Based on these attributes, 230,609 SN entries were successfully linked to the NPI database. The MATCH table, created subsequently, includes the NPI and Record-ID of each match, enabling the identification of corresponding entries in the other tables.

Initially, the new dataset could be used to further standardize attributes such as city and taxonomy. Therefore, the most frequent mismatches of individual attributes in actual matching entries were investigated. This led for instance to the discovery of two thousand entries in the NPI table where the city was given with "phila". These corresponded to SN entries listed under the correct city name "philadelphia". Using this approach, several chronic differences in spelling could be discovered and corrected.

Examination of the table entries reveals potential duplicates in both, the NPI and SN tables. In addition to the standard one-to-one matches, there are two types of special cases. Firstly, some NPIs are matched with multiple Record-IDs from the SN table. Secondly, one Record-ID is linked to multiple NPIs. The primary cause for these scenarios is the presence of residual duplicates in both the NPI and SN tables.

Of all MATCH table entries, 404 have the same Record-ID linked to different NPIs. These cases were sorted by the number of differing column entries in the NPI table. This allows for investigating the NPIs that differ the most but are linked to the same Record-ID. The rationale is that if the most dissimilar NPIs are still duplicates, then more similar NPIs should also be duplicates. Indeed, most highly different NPIs were found to be duplicates. This is not a significant concern for Alpha Sophia. Given the complexity of determining the primary entry and the preference for linking both entries over none, the presence of duplicates in the NPI table is acceptable for this project. However, procedures such as blocking and data splitting must be modified to accommodate duplicates, as outlined in the subsequent steps.

Nevertheless, there are instances where the NPIs were incorrectly matched. The most effective detection method involved checking whether the first letter of the middle name differed. The first letter was selected as many entries only provide this detail for the middle name. In case it differed, at least one of the matches was assumed to be a mismatch. For these 140 cases, all corresponding entries were removed from the MATCH table.

For the reverse scenario, the same approach was applied. 1565 entries in the MATCH table link one NPI to multiple Record-IDs. An analysis of the most disparate cases revealed that there are individuals with multiple profiles, which were active at least once. After eliminating 42 cases where the first letter of the middle name differed, information about the profile picture was utilized to further reduce duplicates. If one of the profiles had a profile picture while the other did not, the profile without a picture was removed. This process resulted in the deletion of 181 entries from the match table.

Following the cleaning process, 230,237 entries remain in the MATCH table. While the disparity in the middle name might not pinpoint all false positives, it is reasonable to assume that the matches are almost exclusively accurate. The intention is to use this dataset as training data for supervised models and as test dataset. However, it is crucial to note that this labeled data is not directly comparable to the unlabeled data, as it lacks a perfect match of first name, last name, and phone number. Consequently, a model trained on the labeled data is likely to perform less effectively on differing data. Therefore, other ways of utilizing this data must be explored.

The objective is to align the labeled data as closely as possible with the unlabeled data. This is essential to allow a generalization of the model performance on the test set to the unlabeled data. While the alignment of names is addressed in Section 4.3, Figure 4.3 displays the distribution of taxonomies and cities by labeling status and dataset.

The examination of the city distribution reveals no significant differences between labeled and unlabeled entries. However, in the case of taxonomies, unlabeled entries have a stronger representation in the "Other" category in both dataframes. As outlined during standardization,

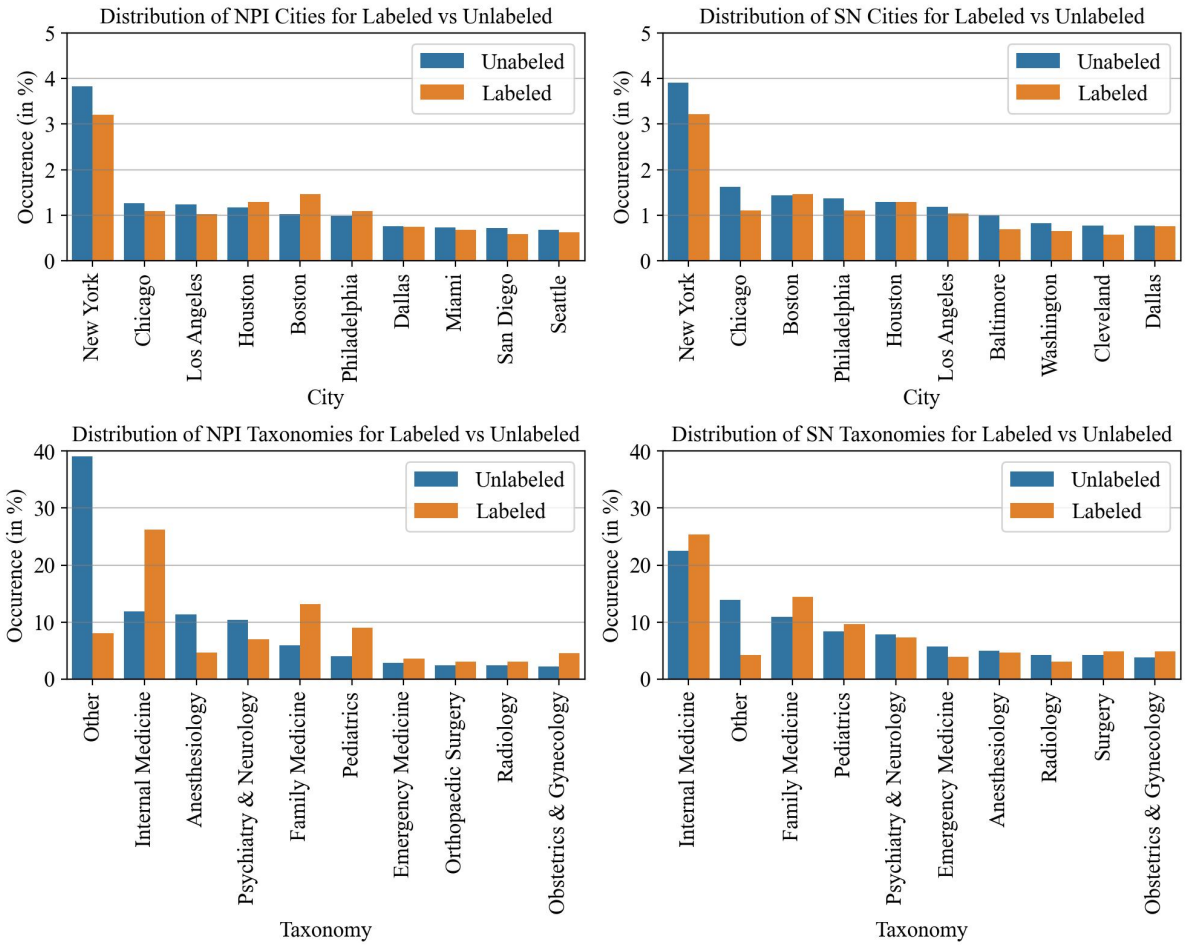

Figure 4.3: Comparison of Distributions of City and Taxonomy by Dataset and Labeling Status

multiple taxonomies were retained in the NPI data, despite their lower representation in the social network. These were primarily categorized under “Other” and "Anesthesiology", leading to an increased proportion of non-matches in the NPI table within these categories. The higher proportion of unlabeled entries within the "Other" taxonomy of the SN data is primarily driven by students who are currently unmatchable due to missing phone numbers. Given these discrepancies in the taxonomy distribution, the use of this attribute should be approached with caution.

4.3 Noise Injection

The training data was created based on a perfect match of first name, last name, and phone number, a condition not met for the unlabeled data. This discrepancy suggests further differences in these attributes. An iterative process of noise injection and modeling, combined with an analysis of the model outputs, revealed that the phone number is a limiting criterion. While

about 90% of the unlabeled data also align on first and last names in case of a match, this figure drops to approximately 12% for the phone number. Due to the high risk of overfitting on the phone number and the limited relevance of this attribute for unlabeled data, the phone number was not considered for modeling purposes. This decision was made to enhance the robustness and generalizability of the model.

Accordingly, the noise injection focused on the name fields. As indicated in Table 4.1, which presents the average word and character count of these fields, the unlabeled data tend to have longer first names in terms of both the number of words and overall length. The difference for middle and last names is less significant. As anticipated, the metrics for labeled entries show no variation between the datasets.

Metric	**Attribute**	**NPI Dataset**		**SN Dataset**	
		Labeled	Unlabeled	Labeled	Unlabeled
Average Number of Words	First Name	1.010	1.032	1.010	1.029
	Middle Name	1.013	1.015	1.013	1.017
	Last Name	1.031	1.032	1.031	1.034
Average Number of Characters	First Name	5.890	6.010	5.890	5.956
	Middle Name	3.276	3.287	3.276	3.291
	Last Name	6.546	6.580	6.546	6.565

Table 4.1: Word and Character Count of Name Attributes

The iterative analysis identified two frequent cases where NPI and SN first names do not match. The most common case is when the first name contains two (rarely more) names instead of one. This occurs in both the NPI and SN databases. In approximately 4% of cases, multiple first names are only present in the NPI table. Conversely, in around 5% of cases, they only appear in the SN table.

To adjust the training data accordingly, a function was developed that selects entries based on given probabilities and appends additional names. For instance, if the NPI first name "peter" is selected, a name is randomly drawn from all the first names in the NPI table with the same gender (e.g., "john") and appended to it (resulting in the new name: "peter john"). In cases where a double name is selected, only the first word is appended. Since the SN table does not provide gender information, the appended name is selected from the NPI table, based on the most common gender associated with the chosen name. As a result of this process, adjustments were made to 9,153 first names of the NPI data and to 11,469 of the first names in the labeled SN data.

The second case occurs less frequently but still encompasses a significant number of instances.

Here, the first name in the SN table is shortened to a single letter. For instance, while the information in the NPI table is "peter", the corresponding entry in the SN data would only be “p”. This situation applies to approximately 0.3% of the data records. Consequently, first names in the SN table were shortened to a single letter at the same ratio, resulting in the modification of 690 entries.

Overall, the differences between labeled and unlabeled data are not substantial and, with few exceptions, can be categorized into two categories. The expectation of uncovering various typographical errors was not confirmed. However, upon closer inspection, this is not surprising. Both the NPI and SN databases are self-maintained and the accuracy of information is of paramount importance. Users typically exert more effort to ensure the correctness of their data on such platforms (Bellatreche et al., 2018).

The creation of labeled data using unique identifying attributes, and subsequent noise injection, led to the generation of a meaningful training and test dataset. This enables the application of supervised models and the possibility to improve results.

4.4 Blocking

As described in Section 2.2, the blocking process aims to reduce the number of necessary comparisons between the two dataframes. It involves a trade-off between reducing the number of comparisons and retaining as many matches as possible. For instance, blocking on the first name (only records with the same name are considered for linkage) can drastically cut down the number of comparisons. However, this approach may filter out records with typographical errors in the first name (e.g., "Petr" instead of "Peter") or cases where two names are given instead of one. Therefore, finding an optimal blocking strategy is essential to ensure accurate and efficient record linkage (Steorts et al., 2014).

In the context of this work’s record linkage problem, another criterion is important to consider: the attributes used for the blocking process should not be the same as those used for the creation of the labeled dataset. For instance, by blocking on phone number, the blocking result for the labeled training data will be different from that of the unlabeled data. In such a scenario, each record in the labeled SN data would have at least one corresponding NPI record with an exact match in first and last name (except for the ones with injected noise), leading to potential overfitting on these attributes. The goal is to maintain as much variability as possible while significantly reducing the necessary comparisons. Furthermore, there should be comparisons that exhibit similarity in different ways while not forming a match.

After exploring various strategies, a deterministic blocking approach was chosen due to its flexibility and lower computational requirements compared to other methods such as clustering. The city and taxonomy attributes were selected as blocking criteria, given their finite number of possible values and the match-likelihood for the same entity. However, these criteria alone were insufficient to reduce the number of comparisons to the required level. Consequently, a third criterion was introduced, requiring that either the first name or the last name match. Although these attributes were already used in the deterministic linkage process, they were only used in combination. It can be assumed that almost every match should coincide in at least one name, independent of the current match status.

The initial blocking step resulted in an excessive removal of matches from the labeled data due to the absence of a blocking partner. This is typically caused by a mismatch in city or taxonomy. While most matches should coincide here, individual details might be stated or maintained differently. To address this, the blocking process was adjusted by adding two additional steps for cases where no blocking partner was found initially. In the second step, blocking was performed only on the city and one name part. The third step required a match of taxonomy, one name part, and the state for all remaining cases. This approach ensured that the different distributions of taxonomy between labeled and unlabeled data, as well as between the datasets, do not overly limit the blocking result. Even if no blocking partner is found in the first step, this might be the case in the second part, regardless of the taxonomy.

The blocking procedure resulted in a dataframe comprising 2,969,809 comparisons. The associated Reduction Ratio is 99.9997%, indicating that the necessary comparisons have been reduced by a factor of over 410 thousand. While the RR pertains to the entire dataset, the Pair Completeness and Pair Quality can only be reliably calculated for the labeled data.

The PC is 95.23%, signifying that more than 95% of the matches are preserved in the dataset, despite blocking. Due to the flexible blocking procedure, it can be assumed that the PC for the unlabeled data will be similar.

The dataframe includes 854,501 labeled comparisons, with 217,919 actual matches, leading to a PQ of 25.50%. Thus, for every labeled match, there are 2.92 labeled non-matches. However, the pair quality cannot be extrapolated to the unlabeled data. For this subset, the PQ is estimated to be lower, around 10%, as only 180-220 thousand matches can be assumed in the 2.1 million unlabeled comparisons. This is attributed to the presence of only 303,507 unlabeled SN records, with 5% filtered out during blocking. Not all of the remaining records will have a corresponding entry in the NPI database. The difference in PQ is mainly caused by more unlabeled entries proceeding to the second blocking stage due to a taxonomy mismatch. As the second stage is more flexible, additional non-matching comparisons are included. Nonetheless, this is not a significant

issue, as a well-trained model should efficiently distinguish these cases, as demonstrated in Section 6.1.

Table 4.2 provides an overview of the performance metrics across all blocking steps, illustrating the trade-off between Reduction Factor (RF) and PC at each stage. The primary objective is to maintain the highest possible PC while reducing the dataset to a size that allows for successful computation. In summary, a significant reduction in the required comparisons has been achieved through blocking. This approach, however, resulted in a loss of approximately 5% of true matches.

Step	**Criteria**	**RF**	**PC (%)**	**PQ (%)**
1	City + Taxonomy + (First Name or Last Name)	712,892	86.97	29.90
2	City + (First Name or Last Name)	527,052	93.04	27.07
3	Taxonomy + State + (First Name or Last Name)	410,859	95.22	25.50

Table 4.2: Blocking Metrics After Each Blocking Step

5 Methodology

Building upon the literature review of probabilistic record linkage conducted in Section 2.2, this chapter outlines the methodologies employed to address Alpha Sophia´s business case. After the description of the implemented models, various performance metrics are introduced. These metrics serve as valuable tools for evaluating the efficacy of the models from a business standpoint.

5.1 Model Choice and Implementation

The record linkage process can be facilitated by numerous supervised and non-supervised models. To address the research question, an assortment of models was employed and compared. Starting with the benchmark, the Fellegi-Sunter model, this section describes the primary types of models utilized and their respective implementations. In addition to the models expounded below, Support Vector Machines (SVM), Naive Bayes, and the Expected Maximization (EM) algorithm were implemented. However, due to their comparatively weak performance, they are not defined and elaborated further. The evaluation and comparative analysis of the individual models are detailed in Section 6.1.

5.1.1 Fellegi-Sunter

The Fellegi-Sunter model is a probabilistic record linkage model that is typically trained unsupervised. It operates with two sets of probabilities, m and u, to calculate weights for each record pair (Fellegi and Sunter, 1969; Linacre, 2023). The m probabilities, or reliability probabilities, represent the likelihood that a specific field matches given the pair is a match. For instance, considering taxonomy, two records describing the same entity might match 70% of the time, leading to an m value of 0.7 for taxonomy. On the other hand, the u probabilities, or commonality, indicate the probability that the attribute will agree if the pair is not a match. A pair of non-matching records may have a 4% chance of matching in taxonomy due to the distinct number of categories, resulting in an u value of 0.04.

The model employs these m and u probabilities to calculate the Bayes Factors (denoted as K), describing the ratio of m to u. The Bayes factor can be interpreted as the change in the likelihood of a match for two records that agree in the given field. For instance, with the values above, the Bayes factor for taxonomy would be $0.7/0.04 = 17.5$. This implies that records with the

same taxonomy are 17.5 times more likely to match than entries that differ. The equations for reliability, commonality, and Bayes factor are displayed below.

$$m = \text{Prob(two values agree | the records do match)} \quad \text{(Reliability)} \tag{5.1}$$

$$u = \text{Prob(two values agree | the records do not match)} \quad \text{(Commonality)} \tag{5.2}$$

$$K = \frac{m}{u} = \frac{\text{Prob(two values agree | the records are a match)}}{\text{Prob(two values agree | the records are not a match)}} \quad \text{(Bayes Factor)} \tag{5.3}$$

The weight of each field is computed using the m and u probabilities and defined as:

$$\text{Weight} = \log_2 \left(\frac{m}{u} \right) \tag{5.4}$$

For each pair of records, these weights are aggregated and combined with a (mostly negative) base weight. If a field agrees, the weight is added, otherwise the negative weight is incorporated. Based on a threshold, the final sum indicates whether a match is predicted.

The Fellegi-Sunter model was implemented using the Splink library in Python, specifically designed for record linkage tasks (Linacre et al., 2022). The process involved using DuckDB, an in-memory analytical database written in C++. An in-memory database stores data in the main memory of the computer to facilitate faster response times. DuckDB enables complex queries and efficient data analysis to be performed within the same process as the application (Raasveldt and Muehleisen, 2023).

The Splink library facilitates the implementation of blocking and provides comparison options for attributes, such as exact match checking or the use of metrics like Jaro-Winkler. However, as the blocking process of the Splink library lacked the possibility of multiple-step blocking, a workaround was employed. The previously blocked dataset was divided into two dataframes, one for the NPI columns and one for the SN columns. A unique ID appended to the original dataset was used as blocking rule, meaning only entries that were already blocked in a row during data preparation are compared with each other.

The Fellegi-Sunter model assumes the independence of the match status of individual attributes. For instance, if the attributes city and address are included, this assumption is violated, as the city match probability increases if the street is the same. Therefore, the city was not considered as feature. Applying different thresholds for a single feature, such as full-name similarity, does not necessarily infringe upon this assumption, given that only the most fulfilled threshold is counted (Daggy et al., 2013; Li et al., 2022). Moreover, the taxonomy was excluded due to its distinct distribution in the training data compared to the unlabeled data.

Consequently, only the full name and address were utilized for comparison. Similarities within these attributes were calculated with various distance metrics. The best performance for address comparisons was achieved by a combination of exact match, Damerau-Levenshtein Distance (with a threshold of <=1), and Jaro-Winkler similarities at thresholds of 0.75 and 0.9. The Damerau-Levenshtein Distance, a variant of the Levenshtein Distance, counts the swapping of two adjacent characters as one operation, instead of two (Damerau, 1964). For name comparisons, the same metrics were applied, but the Jaro-Winkler thresholds were set to 0.5, 0.7, and 0.9.

The thresholds for both metrics determine whether a comparison is considered fulfilled. For the Damerau-Levenshtein distance, this refers to the total number of character insertions, deletions, or replacements. Considering a threshold of <=1, all addresses with a maximum of one change are considered a match. For the Jaro-Winkler metric, the threshold represents a measure of similarity ranging from zero to one.

Once blocking rules and comparisons were specified, the model could be trained. Initially, the u probabilities were estimated based on a sample size of five million, followed by an estimation of the m probabilities. These probabilities are typically estimated using the EM algorithm, a statistical method that iteratively refines parameters to identify the most probable outcomes (Dempster et al., 1977). However, due to the availability of labeled data, the noisy training data could be utilized for training the m probabilities. To allow a performance measurement on the test data, corresponding entries were excluded from the training set.

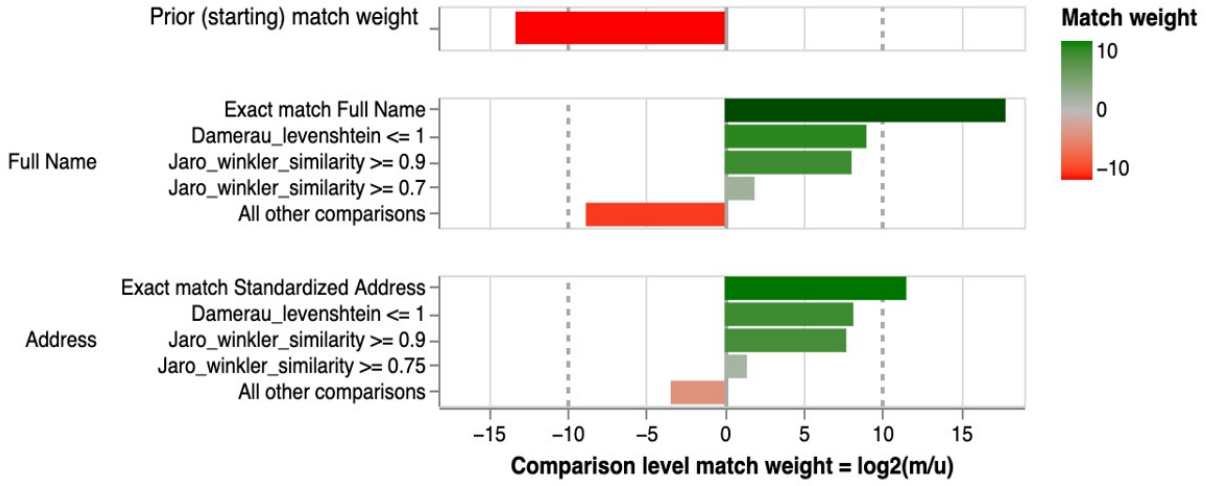

Figure 5.1: Attribute Weights of Fellegi Sunter Model, Adapted from Linacre et al. (2022)

The estimated attribute weights are presented in Figure 5.1. The name comparisons carry a higher overall weight than the comparisons of addresses. Specifically, an exact match in the full-name significantly influences the predicted match probability, while even a single character

discrepancy can alter the outcome. A Jaro-Winkler similarity of below 0.7 has a significant negative influence. This sensitivity is not as pronounced for the address.

The computation between match weights and match statuses of individual attributes yields a match probability for each of the 2.1 million comparisons. Typically, two thresholds are set: one below which a non-match is inferred, and one above which a match is inferred. Values between these thresholds are considered uncertain and require manual inspection. However, to avoid manually checking thousands of potential matches - which contradicts the objective of using a model to bypass manual comparisons - only a threshold for designating a match was set. The classification derived from this process is evaluated in Section 6.1.

5.1.2 Logistic Regression

The logistic regression, a predictive model invented by Berkson in 1944, describes the relationship between a dependent variable and one or multiple explanatory variables (or covariates) (Berkson, 1944). In a conventional regression model, a coefficient is assigned to each explanatory variable that signifies its impact and direction on the dependent variable. The logistic regression employs the inverse logit function to output probabilities between two classes. The logit function maps probability values from zero to one to a range from negative infinity to infinity. It is defined as the logarithm of the likelihood ratio of the probability p of an event:

$$\text{Logit}(p) = \log\left(\frac{p}{1-p}\right), \tag{5.5}$$

where p represents the event probability and $\log$ denotes the natural logarithm.

Using the logit function for a binary classification offers advantages in comparison to a normal regression. It restricts predicted values to the range between zero to one and provides a probabilistic interpretation of the results. Furthermore, it can accommodate non-linear effects, thereby offering greater flexibility and power in modeling complex relationships. By setting a threshold for discrimination, logistic regression becomes a classification model for binary outcomes (Hosmer, 2013). The formulation of the model is:

$$\log\left(\frac{p}{1-p}\right) = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \ldots + \beta_n x_n \tag{5.6}$$

where:

- $\log\left(\frac{p}{1-p}\right)$ is the logit transformation of the positive class probability,
- β_0 is the intercept,
- $\beta_1, \beta_2, \ldots, \beta_n$, are the coefficients of the explanatory variables $x_1, x_2, \ldots, x_n$

The training data created in the previous steps can be used for logistic regression and other supervised models. These models, unlike the Fellegi-Sunter model, were not explicitly designed for record linkage tasks and do not offer the possibility to calculate similarity metrics within the model. This must be done manually. Accordingly, metrics like Jaro-Winkler, Levenshtein, and cosine similarity were calculated before model implementation. To calculate the cosine similarity, the fields are transformed into vectors. Vectorization is the process of converting data into a series of numbers, enabling direct algorithmic comparison (Salton et al., 1975). Cosine similarity, which describes the angle between two vectors, is calculated by dividing the vectors' scalar product by their length (Singhal and Google, 2001). This metric is more suitable for longer text comparisons than metrics like Jaro-Winkler due to its ability to handle high-dimensional data (Wu et al., 2022).

Before implementing a supervised model, a train-test split is essential for effective model evaluation after the training process. A three-way split into training, validation, and testing data is more resistant to overfitting compared to the traditional two-way split (training and testing data) (Harrington, 2018). Thus, the noisy labeled dataset was initially divided into a 20% subset, reserved solely for the final model evaluation. The remaining 80% were further split using the same ratio into training and validation data and can be used during model training.

A key aspect of the split is to ensure entries within a block are in the same training or test dataset. Failure to do so could result in testing the model on data it has already encountered, leading to an inflated test performance (Kapoor and Narayanan, 2022). Therefore, the split was performed on unique Record-IDs and individual blocks were allocated to the subset where the corresponding id is located. However, data leakage can also occur from remaining duplicates in both the SN and the NPI data. To mitigate this, each Record-ID linked to multiple NPIs, and vice versa was exclusively assigned to the training set.

During the implementation of the logistic regression, various hyperparameter combinations and different covariates were tested. The "lbfgs" solver, known for low memory consumption (Liu and Nocedal, 1989), combined with ridge regression, yielded optimal results. Ridge regression is a method to prevent overfitting. The chosen regularization strength is $C = 1$, which implies a balance between model complexity and error tolerance.

Significant covariates included both the Jaro-Winkler distance metrics and the cosine similarities of full-name attributes. The simultaneous use proved effective, as they could compensate for each other's weaknesses. For instance, while the Jaro-Winkler emphasizes deviations at the beginning of the strings, which may not be ideal for name comparisons, the cosine similarity does not consider the positions of discrepancies. For the address, the inclusion of cosine similarities did not enhance performance. Alongside the similarity metrics, a dummy variable accounting

for a match in the middle name boosted the performance. In line with the reasoning provided above, the taxonomy was not included as a covariate in this model.

The model's performance was further enhanced by introducing an interaction term between the cosine similarity of the full name and the length of the (longest) full name in characters. The underlying reasoning is that a high similarity score should carry more weight for longer names. The interaction term was standardized between zero and one to ensure consistency, given the different scales of the variables. As the other covariates exclusively comprised dummy variables and similarity metrics ranging from zero to one, no additional standardization was required.

Figure 5.2 illustrates each covariate's importance. The Jaro-Winkler similarity of the full name is a key attribute, followed by the interaction term. Interestingly, this interaction has a greater influence than the cosine similarity itself. All three name-related attributes hold more weight than the similarity of the address. Despite the middle name dummy having a lower significance, it contributes to the model's explanatory power and is thus retained. The issue of multicollinearity among covariates is tackled in Section 6.3.

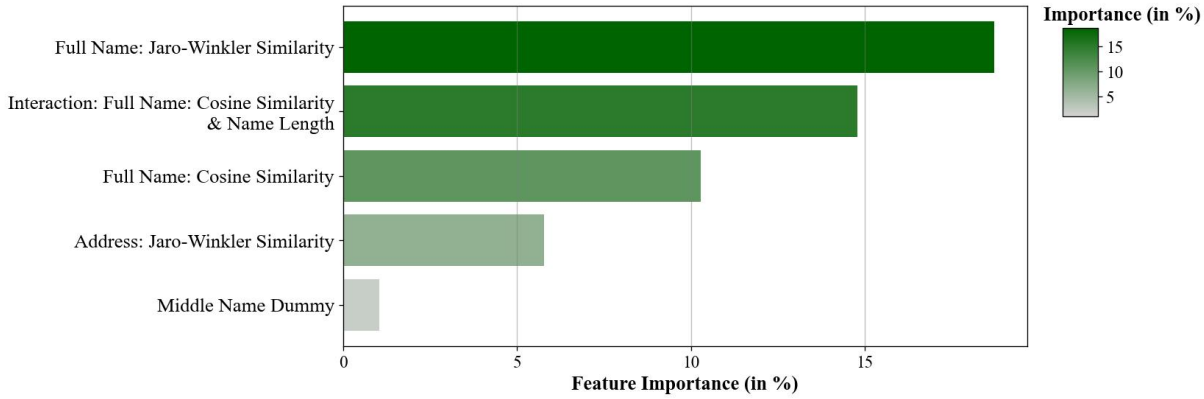

Figure 5.2: Feature Importance of Logistic Regression Model

5.1.3 Random Forest

Random Forest is a machine-learning algorithm that combines multiple decision trees to enhance accuracy (Breiman, 2001). Decision trees have a flowchart-like structure, with each node representing a feature-related condition. Depending on whether the condition is met, a different path is chosen for an observation. At the end of each path, an outcome is predicted, in this case, the match status. By averaging the predictions of each tree, a random forest is created, which reduces overfitting and increases robustness. Figure 5.3 illustrates a simplified decision tree for the business case. However, it is important to note that the final model contains multiple, more complex trees.

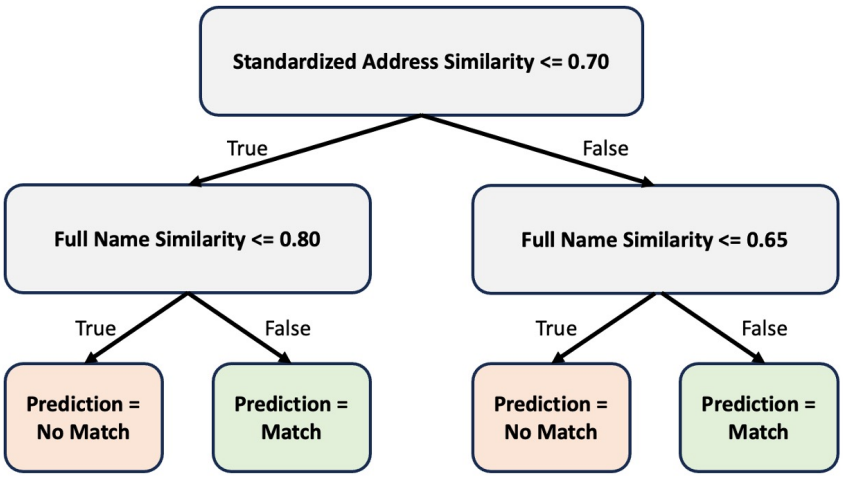

Figure 5.3: Simplified Decision Tree

Given the promising performance of the model, additional techniques like cross-validation were implemented to optimize results. Instead of a simple train-test split, the dataset (excluding the test set) was divided into $k = 5$ subsets. The model underwent five training iterations, each using $k - 1$ folds for training and one for validation. After these iterations, the average of the estimated parameters determined the final model parameters. This method, known as k-fold cross-validation, enhances model robustness and mitigates the risk of overfitting (Rafaeilzadeh et al., 2009).

To further improve the performance of the random forest, hyperparameter tuning was employed. This process involved random search, a method where a grid of hyperparameters is defined, and the model randomly selects combinations for training and evaluation. The advantage of random search is its efficiency: the number of iterations, and resulting computational cost, can be controlled independently. This makes it more efficient in searching the hyperparameter space compared to exhaustive methods like grid search (Scikit-learn, 2023). The F1 score, discussed in Section 5.2, served as the key metric for model optimization during tuning. Table 5.1 provides an overview of the final hyperparameters selected and their roles in the model.

Parameter	Value	Description
n_estimators	250	The number of trees in the forest.
min_samples_split	8	The minimum number of samples required to split a node.
min_samples_leaf	3	The minimum number of samples required to be at a leaf node.
max_depth	40	The maximum depth of the tree.

Table 5.1: Hyperparameters of Random Forest

Besides hyperparameter tuning, feature selection is essential to optimize the random forest model. Similar to the logistic regression, full-name comparisons exhibit the most explanatory power and the interaction of full-name similarity and name length shows high importance. For address comparisons, a selection of both cosine and Jaro-Winkler similarity significantly influences the model. Furthermore, the dummy variable for a match in the middle name tends to slightly improve the model's performance, while the city dummy did not show any improvement. The importance of each feature is elaborated in Figure 5.4.

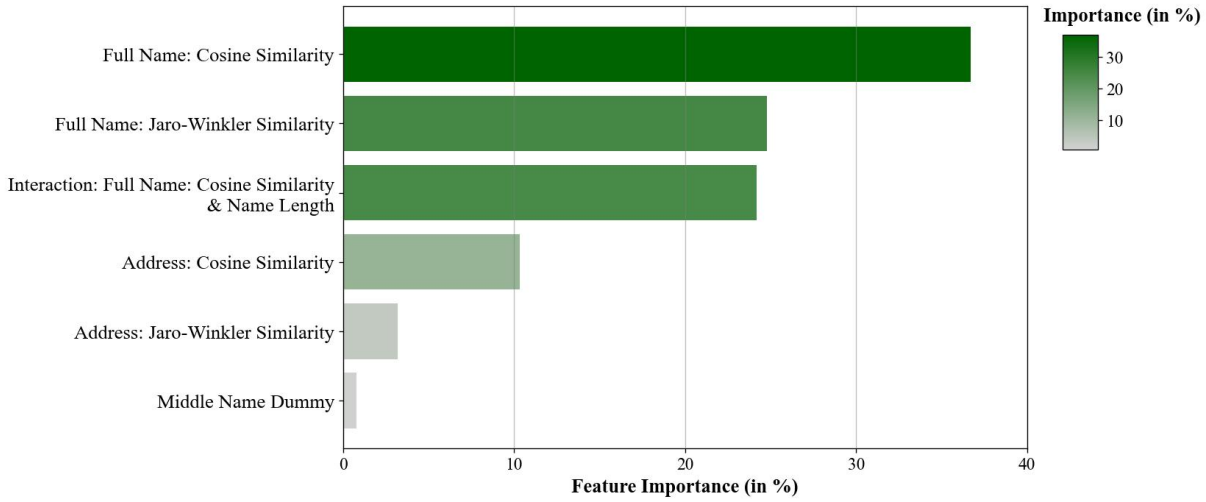

Figure 5.4: Feature Importance of Random Forest Model

5.2 Performance Evaluation Metrics

In a binary classification for record linkage, each prediction can result in one of four outcomes:

- True positive: The prediction and actual class are both "match".
- False positive: The prediction is "match", but the actual class is "no match".
- True negative: The prediction and actual class are both "no match".
- False negative: The prediction is "no match", but the actual class is "match".

The outcomes are depicted in a confusion matrix, as illustrated in Table 5.2.

		Predicted Condition		
		n	**p**	**Total**
Actual Condition	**n′**	True Negative	False Positive	N′
	p′	False Negative	True Positive	P′
	Total	N	P	

Table 5.2: Confusion Matrix for Binary Classification

Numerous metrics exist for the performance evaluation of a classification model. Accuracy represents the proportion of correct classifications considering both true positives and true negatives and is calculated as follows:

$$\text{Accuracy} = \frac{\text{True Positives} + \text{True Negatives}}{\text{Total Number of Predictions}} \tag{5.7}$$

However, accuracy has a significant limitation: it may not be meaningful for imbalanced datasets. For example, considering a disease that occurs in 0.1% of cases. A model predicting everyone as healthy would have an accuracy of 99.9%. Despite this high accuracy, the model would fail to identify any disease, thus completely missing the target.

To address this, other performance metrics are considered. Precision and recall allow a more precise assessment of performance. Precision indicates the frequency with which the model is correct when it makes a positive prediction. Recall quantifies the proportion of actual positive cases that the model correctly identifies. These metrics offer comprehensive insight into the model's performance, especially when dealing with imbalanced datasets (Hand and Christen, 2018). They are defined in the following equations:

$$\text{Precision} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Positives}} \tag{5.8}$$

$$\text{Recall} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Negatives}} \tag{5.9}$$

The relationship between precision and recall further enhances performance evaluation. For instance, the harmonic mean offers a balance between these two measures. This metric is called the F1 score and is calculated as follows:

$$\text{F1 Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{5.10}$$

Despite the usefulness of the F1 score, precision and recall should be assessed independently, especially when false positives and false negatives have different implications (Powers, 2011). This holds true for the presented business case, with more details provided in Section 6.2.

Another method to evaluate a classification model is the Receiver Operating Characteristic (ROC) curve. A graph, illustrating the relationship between the true positive rate (recall) and the false positive rate given as:

$$\text{False Positive Rate} = \frac{\text{False Positives}}{\text{False Positives} + \text{True Negatives}} \tag{5.11}$$

Figure 5.5 illustrates the ROC curve for a well-performing model. The diagonal line represents a model that predicts randomly and serves as a performance baseline. The area filled by the curve above this line indicates the trade-off between the true positive rate and the false positive rate at different thresholds. This area can be quantified and is known as the ROC-AUC score. A ROC-AUC score of 0.5 means randomness, while a score of 1 shows perfect classification (ibid.).

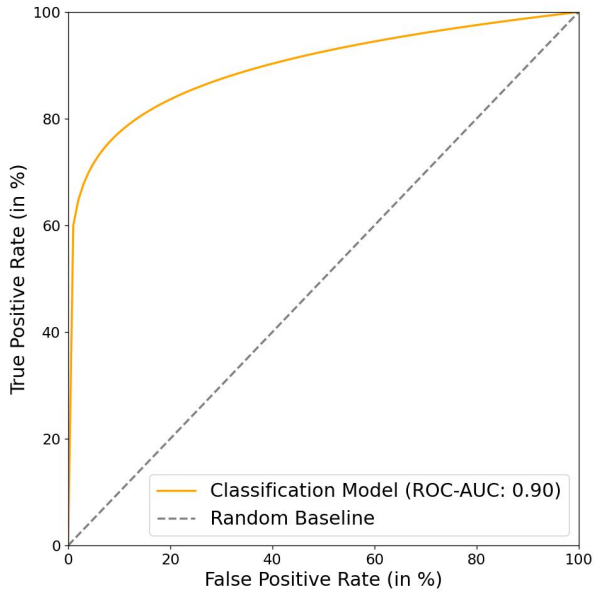

Figure 5.5: Example of the ROC Curve

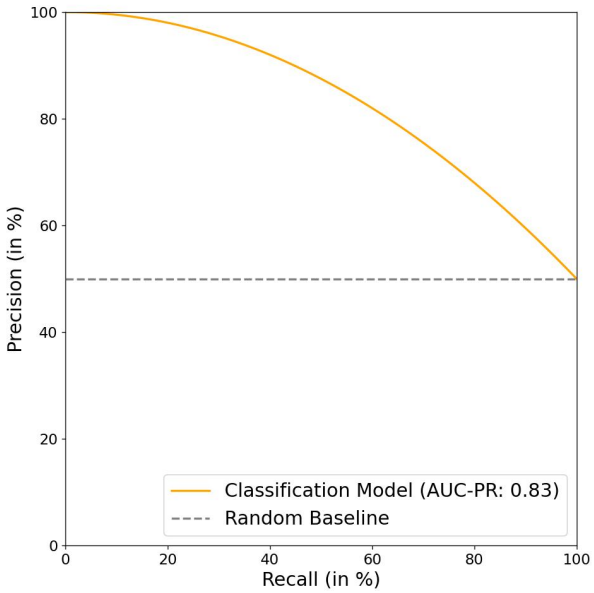

Figure 5.6: Example of the PR Curve

However, for imbalanced datasets, the ROC-AUC score may not accurately reflect the model's performance. Due to a potential bias towards the majority (negative) class, a high number of true negatives is predicted. This results in a low false positive rate, giving the illusion of a well-performing model (Davis and Goadrich, 2006).

Therefore, this work uses the Precision-Recall curve, a more suitable alternative for imbalanced datasets (Tran-The, 2021). This curve plots precision against recall at different thresholds and does not capture true negatives. The AUC-PR score is the area under the curve and can be interpreted similarly to the ROC-AUC value while pertaining only to the positive class. An example of the Precision–Recall curve is illustrated in Figure 5.6. The baseline for a random classifier in a Precision-Recall curve is a horizontal line at the ratio of positives to the total number of samples. A random classifier would, on average, classify correctly at this ratio (Miao and Zhu, 2022). The discussed metrics will be utilized for the performance evaluation in Section 6.1.

6 Results

This chapter encompasses a performance comparison of the implemented models, using the discussed metrics. The comparative analysis facilitates the identification of the most effective model for addressing Alpha Sophia's business problem. Subsequently, an outline of the final implementation of the selected model is provided, offering insights into considerations related to this process. The chapter concludes by presenting potential limitations that could influence the methodology and generalizability of this work, discussing their scope and impact on the results. This critical evaluation not only assures transparency but also paves the way for future research and enhancement.

6.1 Model Comparison

The final evaluation of the models was conducted using a holdout test set. This test set allows the generalization of performance to other unseen data of equal format and type, including the unlabeled data. Each implemented model outputs a score or a class probability in addition to the predicted class, enabling adjustments in class assignments using a threshold.

Figure 6.1 displays the calibration plot of the models. A calibration plot visualizes the alignment of predicted probabilities with actual outcomes and thus helps to assess the reliability of a model's probabilities.

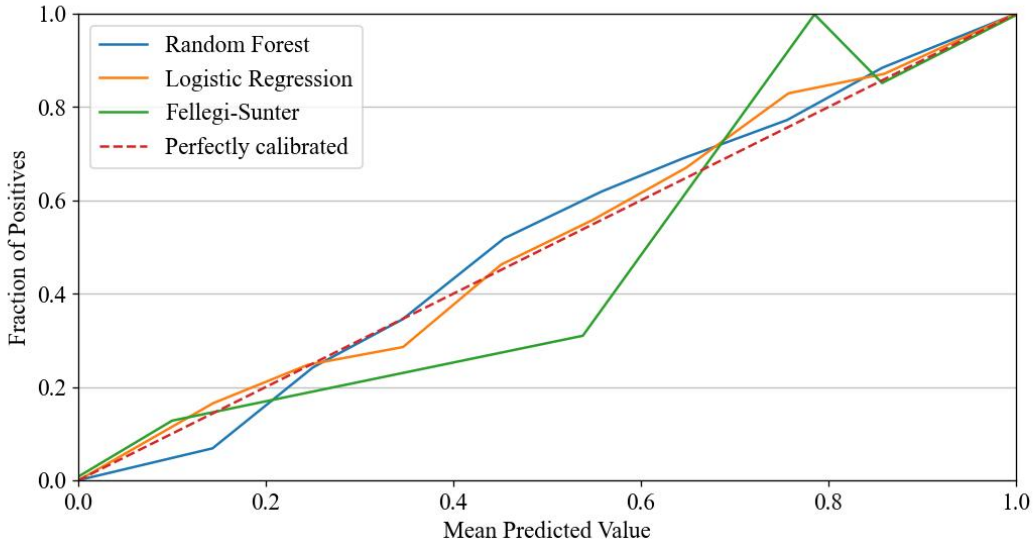

Figure 6.1: Model Calibration Evaluation

The supervised models demonstrate good calibration on the test set. However, there is a slight overconfidence for low-probability matches by the random forest and a minor underconfidence for high-probability matches by both models. In contrast, the Fellegi-Sunter Model presents more extreme discrepancies. This is consistent with the nature of unsupervised algorithms, which lack explicit instructions for class discrimination due to the absence of labeled data (Alloghani et al., 2020).

For Alpha Sophia, the exact probability is not a primary concern, but rather the performance metrics at given thresholds. Considering the satisfactory calibration of the supervised models and the overall underperformance of the Fellegi-Sunter model as displayed below, additional calibration efforts were deemed unnecessary. Nevertheless, it is essential to compare the performance of the models at multiple thresholds to account for different calibrations. Consequently, Table 6.1 displays the confusion matrix and other performance metrics outlined in Section 5.2 at thresholds of 0.25, 0.5, and 0.75.

Upon examining the confusion matrices, it is evident that the random forest model yields the highest number of true positives at each threshold. While the differences compared to the logistic regression are small, there are significant improvements when contrasted with the Fellegi-Sunter model, which predicts a substantial number of false positives. This observation is also mirrored in the calculated metrics, with precision and recall suggesting that the Fellegi-Sunter model faces more difficulties in correctly identifying matches. Considering these metrics, the random forest model outperforms the logistic regression, albeit by a small but consistent margin. Overall, the overview reveals the trade-off between precision and recall based on the chosen threshold.

When considering equal performance weighting across both classes, it is observed that both supervised models deliver their best performance at the 50% threshold. Here, the random forest model achieves an accuracy of 99.84%, signifying that only one in every 625 predictions is incorrect Precision and recall are marginally lower, implying a slightly superior performance of the model on non-matches as on matches. Still, the resulting F1 score of 99.69% indicates a strong performance in predicting matches.

	Threshold 0.25: Model Performance		
Metric	**Fellegi-Sunter**	**Logistic Regression**	**Random Forest**
Confusion Matrix	n: n′ 122,578 / p′ 1,125; p: n′ 4,901 / p′ 41,987	n: n′ 127,149 / p′ 111; p: n′ 330 / p′ 43,001	n: n′ 127,243 / p′ 74; p: n′ 236 / p′ 43,038
Accuracy	96.47%	99.74%	**99.82%**
Precision	89.55%	99.24%	**99.45%**
Recall	97.39%	99.74%	**99.83%**
F-1 Score	93.30%	99.49%	**99.64%**
	Threshold 0.50: Model Performance		
Confusion Matrix	n: n′ 122,994 / p′ 1,186; p: n′ 4,485 / p′ 41,926	n: n′ 127,294 / p′ 183; p: n′ 185 / p′ 42,929	n: n′ 127,344 / p′ 135; p: n′ 135 / p′ 42,977
Accuracy	96.68%	99.78%	**99.84%**
Precision	90.33%	99.57%	**99.69%**
Recall	97.25%	99.58%	**99.69%**
F-1 Score	93.67%	99.57%	**99.69%**
	Threshold 0.75: Model Performance		
Confusion Matrix	n: n′ 127,239 / p′ 3,092; p: n′ 240 / p′ 40,020	n: n′ 127,371 / p′ 332; p: n′ 108 / p′ 42,780	n: n′ 127,398 / p′ 244; p: n′ 81 / p′ 42,868
Accuracy	98.05%	99.74%	**99.81%**
Precision	99.40%	99.75%	**99.81%**
Recall	92.83%	99.23%	**99.43%**
F-1 Score	96.00%	99.49%	**99.62%**

Table 6.1: Model Performance Evaluation at Different Thresholds

In addition to accuracy metrics, examining the Precision-Recall curves provides a more nuanced evaluation. This measure is independent of the threshold and is particularly suitable for the unbalanced dataset. Given the strong performance of the supervised models, a section of the upper right quadrant at values higher than 0.9 is displayed in Figure 6.2 next to the curve. This analysis underscores that the random forest model is the top performer across all thresholds, as it consistently maintains the highest balance between precision and recall.

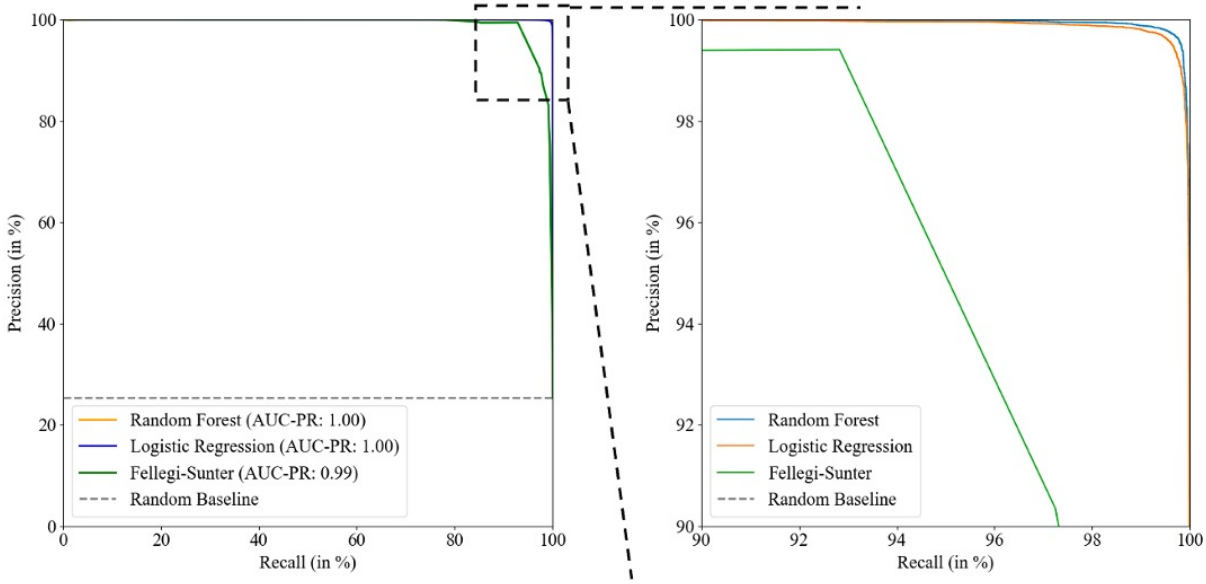

Figure 6.2: Precision-Recall Curve Evaluation

To delve deeper, the performance of the individual models on the manipulated noisy data categories is assessed. The primary performance metrics for each category gathered based on a threshold of 50%, are depicted in Table 6.2.

The models perform slightly better on non-manipulated data compared to the overall performance. When additional name parts are appended to the first name, there is a noticeable decline in the model's performance. This holds true whether the changes were made in the SN or the NPI table, but the impact of changing the NPI name is stronger. In instances where noise has been injected, the models predict with increased caution. This is evidenced by a lower recall across all models, while the precision remains high. The pattern implies that the model only classifies cases as matches if there is a high degree of certainty.

This cautious approach is particularly apparent in the category where the first name has been shortened to one letter. Here, the random forest exhibits the lowest performance among all models. Given this category comprises a relatively small number of cases, the random forest is the overall favored model. However, a significant number of matches in this category will likely go unrecognized.

	Unchanged Data: Model Performance		
Metric	**Fellegi-Sunter**	**Logistic Regression**	**Random Forest**
Accuracy	97.00%	99.84%	**99.89%**
Precision	89.64%	99.54%	**99.69%**
Recall	98.72%	99.80%	**99.88%**
F-1 Score	93.96%	99.67%	**99.78%**
	NPI First Name Attached: Model Performance		
Accuracy	83.99%	97.93%	**98.64%**
Precision	99.77%	**100.00%**	99.93%
Recall	83.23%	97.81%	**98.62%**
F-1 Score	90.75%	98.89%	**99.28%**
	SN First Name Attached: Model Performance		
Accuracy	87.32%	98.34%	**98.84%**
Precision	99.32%	99.80%	**99.86%**
Recall	82.55%	97.83%	**98.49%**
F-1 Score	90.16%	98.81%	**99.17%**
	SN First Shortened: Model Performance		
Accuracy	89.25%	**91.76%**	88.53%
Precision	98.10%	**100.00%**	99.01%
Recall	78.63%	**82.44%**	76.34%
F-1 Score	87.29%	**90.38%**	86.21%

Table 6.2: Model Performance Evaluation for Noise Injection

The results of this analysis are promising despite the reduced recall. From a business perspective, in cases of increased uncertainty, a false negative is preferable to a false positive.

In conclusion, the random forest model demonstrates consistent performance and robustness, indicating its potential efficacy in addressing the business problem. The subsequent section will discuss the implementation and necessary steps to effectively leverage its potential impact.

6.2 Final Model Implementation

Upon finalizing the model, including its hyperparameters and features, it is retrained on all available data. This approach enables the extraction of additional insights from the data previously allocated to the validation and test set. Moreover, it contributes to the model's robustness, thereby enhancing the prediction accuracy on unseen data (Schonlau and Zou, 2020).

After the final training, predictions are made on the unlabeled data, resulting in a match score for each of the 2.1 million comparisons. Since the model does not show perfect calibration, the generated scores cannot be interpreted as probabilities.

The specified threshold for identifying a match significantly influences the implementation of the model on Alpha Sophia's platform. There is a trade-off between the quantity and quality of predicted matches. Minimizing false positives is of utmost importance for Alpha Sophia. Given that information about linked profiles is provided directly to the customer, ensuring the accuracy of this information is crucial. An incorrect link can appear unprofessional and potentially harm reputation and business. Conversely, a missed match, while unfortunate, does not pose a comparable problem in terms of customer satisfaction. Figure 6.3 illustrates the relationship between two key metrics: the number of matches found and the false positive rate, in relation to the selected score-threshold.

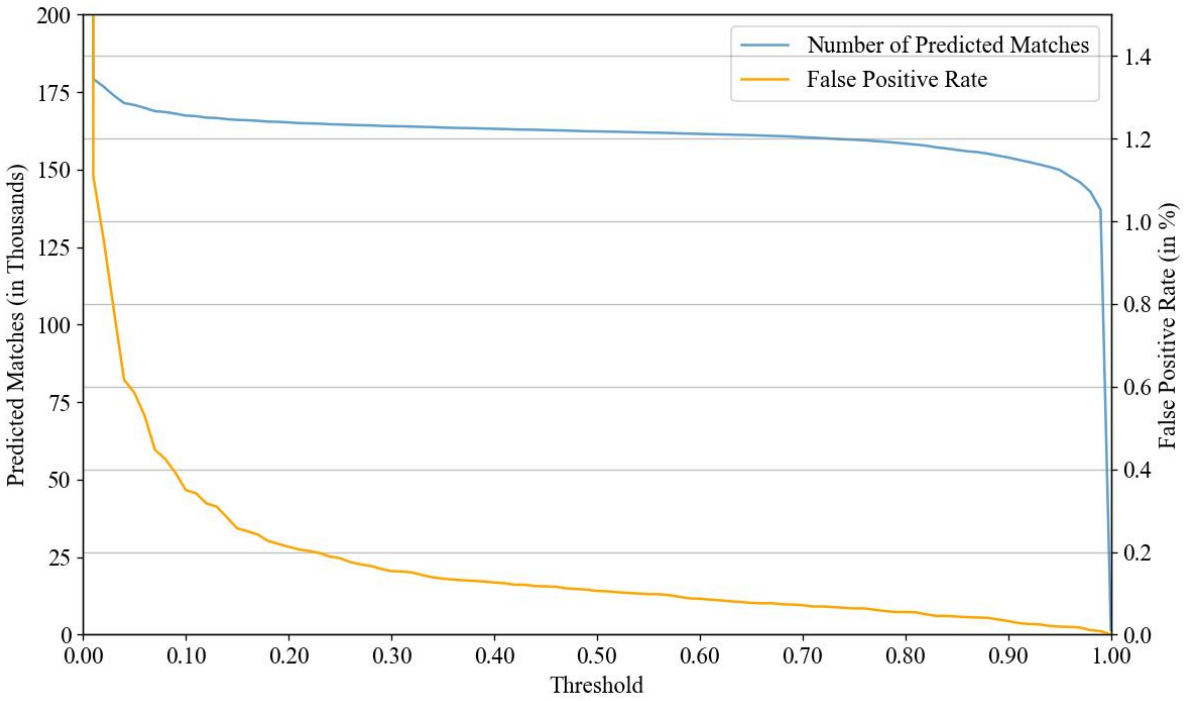

Figure 6.3: Trade-Off: Predicted Matches - False Positives Rate

As the threshold increases, both the anticipated false positive rate and the number of predicted matches decrease. The false positive rate sharply drops to below 0.4%, while the number of predicted matches remains relatively stable at 140-175 thousand. This implies that the proportion of false positives within these reduced predicted matches is relatively high.

Along with Alpha Sophia's preferences, the threshold has been set to a high score of 0.98 for the aforementioned reasons. At this point, 142,914 new matches are predicted, with an anticipated false positive rate of 0.0149%. Based on the test set, only one in 6711 predictions is expected to be a false positive.

Following the implementation of the final model and the preparation of the predictions, the Python notebook was integrated into Alpha Sophia's AWS structure. For this purpose, a streamlined version of the code was implemented, excluding the model comparison and unnecessary steps. This allows Alpha Sophia to internally use, comprehend, and revise the model for new challenges.

6.3 Limitations

Throughout the preparation and modeling process, efforts were made to implement the models in line with their key assumptions. However, due to the nature of the record linkage problem, it was not possible to meet all assumptions. For logistic regression, the primary concerns are the independence of observations and the multicollinearity of features. The lack of observation independence is inherent to record linkage, as comparing an entry from Table A with multiple entries from Table B creates dependent observations (Zhang and Tuoto, 2021). Additionally, multicollinearity is probable, as features like the dummy variable for a middle name match match correlate with the full-name similarity.

Similar challenges arise with the random forest model, which assumes observation independence for unbiased prediction parameters. Multicollinearity can influence the computed feature importance, as it is calculated by the decrease in the model's performance when individual features are omitted (Saarela and Jauhiainen, 2021). If a variable captures the explanatory power of the omitted feature, the importance attributed to the omitted feature can be underestimated.

However, the focus of this work is on generating the best possible predictions, not on interpreting estimators, their importance, or standard error. While it is important to understand how these violations affect the results and their interpretability, the prediction results can still be used to solve the problem (Ayinde et al., 2012). Therefore, these assumption violations were disregarded.

As discussed in Section 5.1.1, the key assumption of the Fellegi-Sunter model is not violated. Yet, the model has a limitation: it only accommodates the match or non-match status of individual attributes at given thresholds and does not process intermediate values (Li et al., 2022). While logistic regression and random forest can handle any similarity value between 0 and 1, the Fellegi-Sunter model needs several thresholds for the same result. Additional m and u values need to be calculated, increasing computational costs. This could be one explanation for the diminished performance.

One limitation of the process for generating labeled data is that some records must be linkable with deterministic methods. The labeled data were created based on a simultaneous match in first name, last name, and telephone number. If the available data does not allow for deterministic linkage, alternative methods must be identified to create the training and test dataset.

Another significant limitation of this work lies in the generalizability. Despite efforts to align the training and test datasets to the unlabeled data through noise injection, not all unique cases and variations were fully captured. Given the vast scale of the data, the focus was primarily on the main differences, potentially overlooking less common variations. This suggests that the performance on the test dataset may not perfectly mirror the performance on the unlabeled data. Measuring this uncertainty presents a challenge without substantial manual effort.

Lastly, the available computational resources did limit the results of this work. For instance, while the blocking rules were adjusted to manage the number of comparisons made by each model, having more resources could allow for the exploration of different blocking rules and procedures. These could include various clustering techniques or relaxing the blocking rules to achieve higher Pair Completeness. Additionally, the selection of hyperparameters for the models was constrained by the available computational resources.

7 Conclusion

The objective of this study was to investigate the potential of probabilistic record linkage techniques in associating entries in the National Provider Identifier (NPI) database with physician's social network profiles, thereby addressing Alpha Sophia's business challenge. An alternative strategy for generating labeled data was developed, followed by the application and comparison of various unsupervised and supervised models. The primary discovery is the superior performance of supervised models, such as the random forest, over the benchmark Fellegi-Sunter when the innovative approach is employed. This finding suggests that supervised models should be favored whenever feasible. Given that this method provides a precise and efficient solution to the challenge, it provides a satisfactory answer to the research question: *"How can probabilistic record linkage methods be applied to effectively and accurately match National Provider Identifier (NPI) records with social network profiles of healthcare providers?"*.

By establishing a high threshold, nearly 143 thousand new matches were identified while maintaining a minimal false positive rate. This corresponds to an increase of approximately 64% in the total number of linked data records. In monetary terms, this equates to a significant manual effort that would incur costs of at least 68,783 euros. Additionally, the immediate implementation of the model resulted in substantial time savings. The methodologies and model presented can be tailored to address other linkage challenges that Alpha Sophia and similar companies encounter.

Consequently, it is recommended that the techniques outlined in this study should be applied in diverse contexts with varying datasets to provide further insights into the generalizability of the performance enhancement. In this process, unique strategies must be explored to deterministically link a portion of the data and effectively examine differences to the non-linkable data. Future research could also investigate methods to quantify the uncertainty between the test set performance and unseen data. This would enable an objective evaluation of the noise injection process and its effectiveness.

Furthermore, it would be beneficial to examine how increased resources could enhance the performance of the blocking and linkage process. The continual advancement of computational power allows for the relaxation of blocking while simultaneously increasing the complexity of the models employed. Despite the presented challenges and limitations, the exploration of this innovative approach - converting an unsupervised problem into a supervised one - unveils new avenues for tackling record linkage problems, underscoring the innovative nature of this work.

Bibliography

Adir, Allon et al. (2022). "Privacy-Preserving Record Linkage Using Local Sensitive Hash and Private Set Intersection". In: *Applied Cryptography and Network Security Workshops*. Ed. by Jianying Zhou et al. Cham: Springer International, pp. 398–424.

AdvaMed (2023). *Medical Device Industry Facts*. <https://www.advamed.org/medical-device-industry-facts/#:~:text=Medical%20technology%20innovators%20are%20committed,create%20medical%20miracles%20every%20day> (visited on 12/04/2023).

Alloghani, Mohamed et al. (2020). "A Systematic Review on Supervised and Unsupervised Machine Learning Algorithms for Data Science". In: *Supervised and Unsupervised Learning for Data Science*. Ed. by Michael W. Berry, Azlinah Mohamed, and Bee W. Yap. Cham: Springer International, pp. 3–21.

Asher, Jana et al. (2020). "Introduction to Probabilistic Record Linkage with a Focus on Linkage Processing for WTC Registries". In: *Int J Environ Res Public Health* 17.18, p. 6937.

Ayinde, Kayode, Emmanuel O Apata, and Oluwayesmisi O. Alaba (2012). "Estimators of Linear Regression Model and Prediction under Some Assumptions Violation". In: *Open Journal of Statistics* 2.5, pp. 534–546.

Bellatreche, Ladjel, Patrick Valduries, and Tadeusz Morzy (2018). "Advances in Databases and Information Systems". In: *Information Systems Frontiers* 20, pp. 1–6.

Berkson, Joseph (1944). "Application of the Logistic Function to Bio-Assay". In: *Journal of the American Statistical Association* 39.227, pp. 357–365.

Breiman, Leo (2001). "Random Forests". In: *Machine Learning* 45, pp. 5–32.

Bruin, Jonathan de (2019). *Python Record Linkage Toolkit: A toolkit for Python Record Linkage Toolkit: A toolkit for record linkage and duplicate detection in Python*. <https://recordlinkage.readthedocs.io/en/latest/about.html> (visited on 12/04/2023).

Cao, Yunbo et al. (2011). "Leveraging Unlabeled Data to Scale Blocking for Record Linkage". In: *Proceedings of the Twenty-Second International Joint Conference on Artificial Intelligence*, pp. 2211–2217.

Chipperfield, James, Noel Hansen, and Peter Rossiter (2018). "Estimating Precision and Recall for Deterministic and Probabilistic Record Linkage: Estimating Precision for Record Linkage". In: *International Statistical Review* 86.2, pp. 219–236.

Christen, Peter (2006). "A Comparison of Personal Name Matching: Techniques and Practical Issues". In: *Sixth IEEE International Conference on Data Mining - Workshops (ICDMW'06)*. Hong Kong: IEEE, pp. 290–294.

Christen, Peter and William E. Winkler (2017). "Record Linkage". In: *Encyclopedia of Machine Learning and Data Mining*. Ed. by Claude Sammut and Geoffrey I. Webb. Boston, MA: Springer US, pp. 1066–1075. DOI: 10.1007/978-1-4899-7687-1_712. <https://doi.org/10.1007/978-1-4899-7687-1_712>.

Committee, National Uniform Claim (2023). *Health Care Provider Taxonomy Code Set*. <https://taxonomy.nucc.org> (visited on 12/04/2023).

Daggy, Joanne K. et al. (2013). "A practical approach for incorporating dependence among fields in probabilistic record linkage". In: *BMC Medical Informatics and Decision Making* 13.97.

Damerau, Fred J. (1964). "A Technique for Computer Detection and Correction of Spelling Errors". In: *Communications of the ACM* 7.3, pp. 171–176.

Davis, Jesse and Mark Goadrich (2006). "The Relationship Between Precision-Recall and ROC Curves". In: *Proceedings of the 23rd International Conference on Machine Learning*. Vol. 6.

Dempster, A.P., N.M. Laird, and Rubin D.B. (1977). "Maximum Likelihood from Incomplete Data Via the EM Algorithm". In: *Journal of the Royal Statistical Society: Series B (Methodological)* 39.1, pp. 1–22.

Dunn, Halbert L. (1946). "Record Linkage". In: *Am J Public Health Nations Health* 36.12, pp. 1412–1416.

Elfeky, M.G., V.S. Verykios, and A.K. Elmagarmid (2002). "TAILOR: a record linkage toolbox". In: *Proceedings 18th International Conference on Data Engineering*. San Jose, CA, pp. 17–28.

Enamorado, Ted, Benjamin Fifield, and Imai Kosuke (2017). *fastLink: Fast Probabilistic Record Linkage with Missing Data. Version 0.6.* <https://github.com/kosukeimai/fastLink> (visited on 12/04/2023).

Fellegi, Ivan P. and Alan B. Sunter (1969). "A Theory for Record Linkage". In: *Journal of the American Statistical Association* 64.328, pp. 1183–1210.

Fienberg, Stephen E. and Daniel Manrique-Vallier (2009). "Integrated methodology for multiple systems estimation and record linkage using a missing data formulation". In: *Advances in Statistical Analysis* 93, pp. 49–60.

Fleming, Michael, Brad Kirby, and Kay I. Penny (2012). "Record linkage in Scotland and its applications to health research". In: *Journal of clinical nursing* 21.19-20, pp. 2711–2721.

Gelman, Andrew et al. (2013). *Bayesian Data Analysis*. 3rd ed. NY: Chapman and Hall/CRC.

Gkoulalas-Divanis, Aris et al. (2021). "Modern Privacy-Preserving Record Linkage Techniques: An Overview". In: *IEEE Transactions on Information Forensics and Security* 16, pp. 4966–4987.

Hand, David and Peter Christen (2018). "A note on using the F-measure for evaluating record linkage algorithms". In: *Statistics and Computing* 28.3, pp. 539–547.

Handelsblatt (2023). *Brutto-Netto-Rechner 2023*. <https://www.handelsblatt.com/brutto-netto-rechner/> (visited on 12/04/2023).

Harrington, Peter B. (2018). "Multiple versus single set validation of multivariate models to avoid mistakes". In: *Critical reviews in analytical chemistry* 48.1, pp. 33–46.

Harrison, Liz et al. (2021). *B2B sales: Omnichannel everywhere, every time*. <https://www.mckinsey.com/capabilities/growth-marketing-and-sales/our-insights/b2b-sales-omnichannel-everywhere-every-time#/> (visited on 12/04/2023).

Hosmer, David W. (2013). *Applied Logistic Regression*. 3rd ed. NY: John Wiley and Sons, Inc.

Jaro, Matthew A. (1976). *UNIMATCH: A Record Linkage System: User's Manual*. Tech. rep. Washington D.C., WA: U.S. Bureau of the Census, Washington, D.C.

Kapoor, Sayash and Arvind Narayanan (2022). "Leakage and the reproducibility crisis in machine-learning-based science".

Kevin, O'Hare, Anna Jurek-Loughrey, and Cassio de Campos (2019). "A Review of Unsupervised and Semi-supervised Blocking Methods for Record Linkage". In: *Linking and Mining Heterogeneous and Multi-view Data*. Ed. by Deepak Padmanabhan and Anna Jurek-Loughrey. Cham: Springer, pp. 79–105. DOI: 10.1007/978-3-030-01872-6_4.

Khan, Shahidul I., Abir B.A. Khan, and Abu Sayed L. Hoque (2022). "Privacy Preserved Incremental Record Linkage". In: *Journal of Big Data* 9, p. 105.

Kilss, Beth and Wendy Alvey (1985). "Record Linkage Techniques". In: *Proceedings of the Workshop on Exact Matching Methodologies*. Arlington, VA: Internval Revenue Service Publication.

Li, Xiaochun, Huiping Xu, and Shaun Grannis (2022). "The Data-Adaptive Fellegi-Sunter Model for Probabilistic Record Linkage: Algorithm Development and Validation for Incorporating Missing Data and Field Selection". In: *Journal of Medical Internet Research* 24.9, p. 33775.

Linacre, Robin (2023). *m and u probabilities in the Fellegi-Sunter model*. <https://www.robinlinacre.com/m_and_u_values/> (visited on 12/04/2023).

Linacre, Robin et al. (2022). "Splink: Free software for probabilistic record linkage at scale." In: *International Journal of Population Data Science* 7.3, p. 23.

Liu, Dong C. and Jorge Nocedal (1989). "On the limited memory BFGS method for large scale optimization". In: *Mathematical Programming* 45, pp. 503–528.

Medicare, Centers for and Medicaid Services (2023). *NPPES NPI Registry*. <https://npiregistry.cms.hhs.gov/search> (visited on 12/04/2023).

Miao, Jiaju and Wei Zhu (2022). "Precision–recall curve (PRC) classification trees". In: *Evolutionary Intelligence* 15, pp. 1545–1569.

Navarro, Gonzalo (2001). "A Guided Tour to Approximate String Matching". In: *Association for Computing Machinery* 33.1, pp. 31–88.

Newcombe, Howard B. (1967). "Record Linking: The Design of Efficient Systems for Linking Records into Individual and Family Histories". In: *American Journal of Human Genetics* 19.3 PT 1, pp. 335–359.

Nowak, Robert et al. (2021). "Record Linkage of Chinese Patent Inventors and Authors of Scientific Articles". In: *Applied Science* 11.18, p. 8417.

Pollock, Joseph J. and Antonio Zamora (1984). "Automatic Spelling Correction in Scientific and Scholarly Text". In: *Communications of the ACM* 27.4, pp. 358–368.

Powers, David M. W. (2011). "Evaluation: from precision, recall and F-measure to ROC, Informedness, Markedness and Correlationeasure to ROC, informedness, markedness and correlation". In: *nternational Journal of Machine Learning Technology* 2.1, pp. 37–63.

Raasveldt, Mark and Hannes Muehleisen (2023). *DuckDB*. <https://github.com/duckdb/duckdb> (visited on 12/04/2023).

Rafaeilzadeh, Payam, Lei Tang, and Hang Liu (2009). "Cross-Validation". In: *Encyclopedia of Database Systems*. Ed. by Ling Liu and M. Tamer Özsu. Boston, MA: Springer, pp. 532–538.

Roos, Leslie L. and Andre Wajda (1991). "Record Linkage Strategies". In: *Methods of Information in Medicine* 30.02, pp. 117–123.

Saarela, Mirka and Susanne Jauhiainen (2021). "Comparison of feature importance measures as explanations for classification models". In: *SN Applied Science* 3.272.

Salton, G., A. Wong, and C.S. Yang (1975). "A vector space model for automatic indexing". In: *Communications of the ACM* 18.11, pp. 613–620.

Schonlau, Matthias and Rosie Y. Zou (2020). "The random forest algorithm for statistical learning". In: *The Stata Journal* 20.1, pp. 3–29.

Scikit-learn (2023). *Comparing randomized search and grid search for hyperparameter estimation*. <https://scikit-learn.org/stable/auto_examples/model_selection/plot_randomized_search.html#> (visited on 12/04/2023).

Singhal, Amit and I. Google (2001). "Modern Information Retrieval: A Brief Overview". In: *IEEE Data Engineering Bulletin* 24, pp. 1–9.

Steorts, Rebecca C. et al. (2014). "A Comparison of Blocking Methods for Record Linkage". In: *Privacy in Statistical Databases*. Ed. by Josep Domingo-Ferrer. Cham: Springer International Publishing, pp. 253–268.

Taylor, Ian, Andee Kaplan, and Brenda Betancourt (2023). "Fast Bayesian Record Linkage for Streaming Data Contexts". In: *Journal of Computational and Graphical Statistics*.

Tran-The, Tam D. (2021). *Precision-Recall Curve is More Informative than ROC in Imbalanced Data: Napkin Math and More*. <https://towardsdatascience.com/precision-recall-curve-is-more-informative-than-roc-in-imbalanced-data-4c95250242f6> (visited on 12/04/2023).

Vatsalan, Dinusha, Dimitrios Karapiperis, and Vassilios .S. Verykios (2022). "Privacy-Preserving Record Linkage". In: *Encyclopedia of Big Data Technologies*. Ed. by Sherif Sakr and Albert Zomaya. Living Edition. Cham: Springer, pp. 1–8.

Vatsalan, Dinusha, Ziad Sehili, et al. (2017). "Privacy-Preserving Record Linkage for Big Data: Current Approaches and Research Challenges". In: *Handbook of Big Data Technologies*. Ed. by Albert Y. Zomaya and Sherif Sakr. Cham: Springer, pp. 851–895.

Winkler, William E. (2014). "Matching and record linkage". In: *WIREs Computational Statistics* 6.5, pp. 313–325.

Wu, Zhengjiang et al. (2022). "A Distributed Attribute Reduction Algorithm for High-Dimensional Data under the Spark Framework". In: *International Journal of Computational Intelligence Systems* 15.22.

Zhang, Li-Chun and Tiziana Tuoto (2021). "Linkage-Data Linear Regression". In: *Journal of the Royal Statistical Society Series A: Statistics in Society* 184.2, pp. 522–547.