
# EMPLOYING PROBABILISTIC MATCHING ALGORITHMS FOR IDENTITY MANAGEMENT IN THE TELECOMMUNICATION INDUSTRY

**Odedina, Omolade Temitope**

A Thesis submitted to the Faculty of Computer Science at the

African University of Science and Technology

In Partial Fulfilment of the Requirements for the degree of

Master of Science in the Computer Science Department.

June 2019

African University of Science and Technology [AUST]

*Knowledge is Freedom*

# APPROVAL BY

**Supervisor**

Surname: EKPE

First name: Okorafor

Signature:  21st June 2019

**The Head of Department**

Surname: DAVID

First name: Amos

Signature:

# ABSTRACT

*The telecommunication industry has a lot of data related to households, individuals and devices. Advertisers pay a premium to ensure they advertise to their target audience. To ensure that content is personalized, it is necessary to accurately predict who is using a device in real time. A probabilistic matching algorithm to determine the profile of an individual based on behavioural analytics is developed and implemented. Two datasets 'People data' and 'Device data' were linked and matched using social behaviours exhibited by individuals whose information are contained in the People data and by devices whose addresses show specific social behaviours of individuals who use the devices. A match score was generated to show the accuracy of a pair of records from the different datasets (i.e. to show if both records are indeed a match or not).*

**Key words: Probabilistic Matching Algorithms, Social Behaviour, Telecommunication Industry, Match Score.**

# DEDICATION

I dedicate this work to my parents, Engineer and Mrs Kayode Odedina, to my brothers, Odedina Damilola and Odedina Okikioluwa and to myself.

# ACKNOWLEDGEMENT

I will show my appreciation first, to God Almighty, for giving me the grace to complete this project.

In addition, I want to express my gratitude to my parents, Engineer and Mrs Odedina for their unending love and support. Thank you for always being there for me while growing up, and for teaching me.

I will to appreciate my supervisor, Dr. Ekpe Okorafor who has taken his time to be diligent in his work. The guidance, mentorship and support are appreciated in so many ways. I also thank the staff and faculties of Computer science who have impacted knowledge in me.

More so, I will also like to thank my classmates for their support and help all round. Thank you everybody.

# TABLE OF CONTENTS

# LIST OF FIGURES



# LIST OF TABLES

# CHAPTER ONE

## 1.1. INTRODUCTION

The Telecommunications industry is one of the subsectors that make up the Information and Telecommunication Technology sector. This industry includes all telephone companies, Internet Service Providers (ISP), radio companies and television companies. The Telecommunication industry gets wider and more complex due to the proliferative nature of the devices involved.

The Telecommunication industry is a very high revenue generating company. Research has it, that due to the increasing scope of the Telecommunication industry, telecommunications service revenue will grow from $2.2 trillion in 2015 to $2.4 trillion in 2019. A way to achieve this is through advertisement. Advertisers pay huge amount of money to advertise their services. So, there is the need to advertise products and services and more so, to advertise to the target user. When products and services are advertised to the target audience, there is higher chance of companies selling and users purchasing. Therefore, there is a need to know who uses a device at a particular time and what such user is interested in. This rapid increase in the number of devices available allows individuals (or households, as the case may be) to own more than one device. The need for the identification of users per time and also for advertisement to target audience is where **Identity Management** is taken into consideration.

Since Identity Management deals with individuals, different attributes of individuals are used to implement it. Attributes of individuals are classified into personal attributes, social behavior attributes and social relationship attributes (Li & Wang, 2015). To carry out efficient Identity Management, attributes and individuals themselves must be accurately matched. Hence, the use of a matching algorithm for best match.

## 1.2. BACKGROUND STUDY

### 1.2.1. On Record Linkage

The term record linkage which was introduced by Halbert L. Dunn through his paper "Record Linkage" published in 1946, was referred to as the linking of medical records associated with individuals. Halbert Dunn described a system developed by the Dominion Bureau of Statistics in Canada for which information containing names of individual from microfiche was put on punch cards and after this, lists were printed for verification and review by different agencies in Canada. The methods above were cost-effective at the time because they were far more efficient than purely manual matching and maintenance of paper files.

Generally, computerised record linkage began with methods introduced by geneticist Howard Newcombe (in his papers Automatic Linkage of Vital Records published in 1956 and Record Linkage: Making Maximum use of the discrimination power of identifying information published in 1962) who used odds ratios and value-specific frequencies (for example common value of last name 'Smith' has less distinguishing power than rare value 'Zabrinsky'). Then Fellegi and Sunter (in their 1969 paper, A Theory for Record Linkage) gave a mathematical formalisation of Newcombe's ideas. They proved the optimality of the classification rule of Newcombe and introduced many ideas about estimating 'optimal' parameters (probabilities used in the likelihood ratios) without training data. Training data, which makes suitable parameter estimation much easier, is a set of record pairs for which the true matching status is known, created, for example, through certain iterative review methods in which 'true' matching status is obtained for large subsets of pairs (Winkler, 2015).

### 1.2.2. On Identity Management

Due to the ubiquitous nature and the rapid rate of development of the technology and web applications world, access to different applications are made quite easy for illegal users. The developers are then driven to create more secure environments for applications by allowing for more careful control.

Identity Management is dated as far back as the 19th century where in 1853, the government of the United Kingdom made it compulsory for citizens to register new births and by 1902, the entire United States was standardized. In the 20th century, in the united states, the first driver's license , the first passport, the first Social Security Number, the first digital identities and passwords and commercial internet was born. The use of passwords was introduced to keep the information about individuals and bodies private. In those times, Identity management was generally made up of manual sheets and other services used to track accounts. As soon as commercial internet was born, Traditional Identity Management systems were adapted for online applications.

In the year 2000, the population of internet users grew to about 400 million people used and this increased the vices, such as identity theft, performed by and on people through the internet. Due to the need to stop these vices, an effective and efficient system was developed and Identity Management Stack was birthed. This stack system had a limitation though – it was very expensive to maintain

In 2010, Identity as a Service cloud was created with the aim of simplifying, automating and reducing costs associated with the Stack. From 2010 till date, Identity Management has been fully digitized and is in successful use in today's computing.

## 1.3. LIMITATIONS OF SOME WORKS

Although, extensive research has been going on as regards Identity Management, most works have focused solely on the personal attributes of individuals and paid little or no attention to behavioural attributes of individuals (Li & Wang, 2015). Some works also dwelled on just supervised and semi-supervised technique of machine learning for probabilistic linkage(Diaz-Morales, 2015).

It has also been observed that probabilistic matching has not been applied to the telecommunication industry to a large extent. This work attempts to explore and implement probabilistic matching algorithms on data from telecommunication industries.

## 1.4. PROBLEM STATEMENT

New products and services are being thought of, designed, implemented and released frequently and such services can best reach individuals through advertisement. The fastest forms of advertisement are those that are done through telecommunication devices. The audience and can see and hear about what is being advertised billions of miles away. It is one thing to advertise to the public, it is another thing to advertise to the target individual. If the target individual is not reached, sales of such services will be low which will result to low profit or even loss for the company doing the advertisement and for the telecommunication industry at large. Also, for identity management, the use of both personal attributes and behavioural attributes of an individual should be taken into consideration. Also, all techniques of machine learning should be incorporated and the technique that produces the best match should be noted.

## 1.5. AIM OF STUDY

The aim of this study is to develop a probabilistic matching algorithm to link individuals to devices hence, knowing the profile of an individual (which includes individual's preferences and interests) and advertising most suitable products to such individuals.

## 1.6. OBJECTIVES

**I.** Collate data, perform data quality assessment and data cleansing.

**II.** Group attributes of individuals into Personal attributes and Social (behavioural) attributes.

**III.** Use Machine Learning Techniques to develop a probabilistic matching algorithm to determine the profile of an individual (Identity Management) based on behavioural analytics.

## 1.7. TECHNOLOGIES REQUIRED

### a) Machine Learning:

Machine learning is usually seen as a subset of Artificial Intelligence and it is defined as a the scientific study of algorithms, statistical models and other features that computer systems can use so that, without little or no human intervention, they can perform their tasks, relying on patterns and inferences instead. Algorithms developed in Machine Learning usually build a model using a sample data that is generally referred to as the training data. Analysis from machine learning can be predictive, exploratory, descriptive or prescriptive – each analysis, largely dependent on the Machine Learning Technique used.

**b) Supervised Learning and Unsupervised Learning**:

Machine learning is divided generally into Supervised Learning and Unsupervised Learning. In supervised learning, there is a particular output that is needed to be gotten from the system and so the input is manipulated and worked on until such output is gotten. For unsupervised learning, no specified input is given to the system. The system receives input data and produces an output data. Inference and deductions are drawn from the output of the system.

**c) Data Analysis:**

Data Analysis, sometimes referred to as Data Modelling and having many approaches to it, is a process of inspecting, cleansing, transforming and modelling data with the aim of discovering useful information and supporting decision making. Descriptive Statistics, Exploratory Data Analysis (EDA) and Confirmatory Data Analysis (CDA) are the three main classification of Data Analysis.

**d) Probabilistic Matching:**

It is also known as Fuzzy matching or record linkage. It is the task of finding records in dataset and such that this records refer to the same entity or in this case, individuals even if the data are from different sources. It takes into account a wider range of potential data identifiers, computes weights for each identifier based on the predicted ability for this identifier to identify a match and then uses the computed weight to check if a record pair refer to the same entity.

## 1.8. DEFINITION OF TERMS

**Advertisement:**

A marketing communication that makes use of an openly sponsored non-personal message to create, develop, promote and sell a product or a service. Advertisements

can be done through various communication media outlets such as newspapers, magazines, radio, television, blogs, social media etc. Advertisement can be classified by style, target audience, geographic scope and purpose.

**Identity Management:**

It is basically the authorization and authentication of a user to grant the user access to resources made available. It also determines to an extent, what the user will do with such resources. In Identity Management, some key areas are to be undergone in order to ensure accuracy. The areas include Directory services, Identity Administration and Access Management. Directory Services allow files and resources to be located and also allows for access of user data. Identity Administration monitors the lifestyle of the data i.e. how often data changes and also effects the changes. Access Management, used interchangeably with Identity Management. It authorizes and authenticates (verifies and validates) users of a particular resource by accessing their data and deeming them fit to access the data or not (Oracle Corporation, 2008).

**Probabilistic Matching Algorithms:**

These are algorithms which are developed and implanted in order to make use of record linkage (probabilistic matching) to work on data such that the data needs less human intervention.

## 1.9. GENERAL PROBLEMS OF PROBABILISTIC MATCHING

i. There is no personal or key identifier on one or both datasets to be matched.
ii. There could be the problem of missing data in the dataset(s).
iii. Some information such as gender, address, state etc. could be made available for matching.

iv. Some Identifiers have more weight or discriminatory powers than other identifiers.

v. Maximum and minimum threshold to determine matching state of identifiers are not easily known.

## 1.10. PROJECT SCOPE:

**Chapter 1**: This covers the introduction, background study, the problem statement of the project, the aim and the objectives of the work. It also covers the definition of terms, the technologies required and the general problems of probabilistic matching.

**Chapter 2**: This chapter contains the literature review of related works to this project and through the review, show how this project stands in its own unique way and how past works can be of great help in achieving the aims of this project.

**Chapter 3:** In this chapter, the materials and methods used for the project are discussed and implemented. The python programming language was used to prepare codes that were used to implement this project. Different python libraries were used to carry out some specific processes

**Chapter 4:** The testing and validation of the project was analyzed in this chapter. The testing of the project was divided into three parts; the data matching itself, the comparison of string comparators on the set of matched and unmatched data and lastly, comparing and confirming matching score and Positive Predictive Values gotten from the Fuzzywuzzy record matching, manual reviews and EM (Expectation Maximization) Algorithms.

**Chapter 5:** Conclusions, contributions, challenges, recommendations and future works to be done comprises this chapter.

# CHAPTER 2

## 2.1. RECORD LINKAGE

It is the task of finding records in a data set that refers to the same entity across various data sources. There are two major methods for record linkage.

### 2.1.1. Deterministic Record Linkage Versus Probabilistic Record Linkage

Deterministic Record Linkage is a record linkage technique that gives an accurate result only when the data is of high quality, with little or no noise or error. The major problem researchers encounter in deterministic record linkage is that it does not take into account discriminatory power of different identifiers. It assigns unit weight to all identifiers regardless.

Probabilistic Record Linkage is very efficient for messy and noisy datasets. It is a record linkage technique that takes into account the discriminatory power of identifiers and assigns weight to the identifiers based on the level of discriminatory power.

## 2.2. IDENTITY MANAGEMENT

It is basically the authorization and authentication of a user to grant the user access to resources made available.

### 2.2.1. Authorization Versus Authentication

Two terms that are associated directly with identity management are Authorization and Authentication.

Authorization is simply referred to as Access Policy. It is the process of granting individuals the access or privileges to resources or applications in a system. Authentication is the approval or confirmation of the identity of an individual or entity.

## 2.3. RELATED WORKS

### A Probabilistic Matching Algorithm for Computer Vision:

Camps et al., (1994) developed a model based vision system that will be used to find correspondences between the features an object model has and also between the features of the image model. The purpose of this vision system is to recognize, localize or inspect these objects and their corresponding images. In this paper, the heuristic method and the relational matching algorithms were considered. The paper posed the relational matching problem as a special case of the pattern recognition problem and therefore proposed a probabilistic model to describe the images of an object. The algorithm proposed in this paper keeps the size of the problem being worked on under control by efficiently reducing the search space, even as feature matching naturally is exponential**.**

### An Efficient Algorithm for Fingerprint Matching:

Wang et al., (2006) proposed a novel topology-based algorithm for fingerprint matching. The aim of the paper was to develop this novel topology-based algorithm to be effective and more efficient than other fingerprint algorithms and also to improve the speed and accuracy of fingerprint identification. This technique is referred to as Delaunay triangulation based technique. Three major aspects of fingerprints were considered in this paper namely the local matching, tolerance deformation and global matching. Computational geometry methods such as Delaunay triangulation and spatial interpolation are used. Delaunay triangle edges instead of whole minutiae triangle edges for the choice of the matching index because Delaunay triangles only accommodate points in a set that do not have corresponding points in other sets. Some elastic deformations are hazardous to fingerprint verification and to mitigate

these deformations, a model referred to as Radial Basis Function (RBF) was introduced. A maximum bipartite scheme was introduced to improve matching accuracy. To evaluate biometrics information, two criterions namely False Accepted Rate (FAR) and the False Rejected Rate (FRR) were used.

**Identity Matching Based on Probabilistic Relational Models:**

Li et al., (2006) proposed a probability relational model based approach to enhance identity matching by matching these identities in databases. According to the paper, when the work was based solely on personal attributes, an average precision of 53.73% was achieved. When social activity attributes were introduced, precision increased to 54.64% and when social relationship attribute was included, precision increased to 68.27%. This research discussed on how the features that identify people's social relationship and activities could be derived for the purpose of identity matching and also, how the social status of an individual can improve the performance identity matching. This work showed the difference between standard data mining and relational learning. While standard data mining works with only flats i.e. single tables, relational learning can draw its data from multiple tables that are related, in a database structure.

**Identity Matching Using Personal and Social Identity Features:**

Jiexun et al., (2011) aimed to develop data mining objectives that could match identities referring to the same person. They discovered that identity matching techniques which are in existence dwell mainly on personal features of individuals without taking into consideration, the social features and social behaviour of such individuals. The paper proposes a new technique that takes into consideration, both

the personal identity of an individual and the social identity of such individual. The technique was built upon a probabilistic relational model that used a relational database structure to extract social identity features. The paper discussed heuristic and machine learning approaches to identity matching. It also analyzed the advantages the machine learning approach had over the heuristic approach. This paper focused on criminal activities, precisely using identity matching to identify current terrorists and potential terrorists.

**A frame work of Identity Resolution: Evaluating Identity Attributes and Matching Algorithms:**

Li and Wang (2015) worked on a project whose aim was to overview various identity attributes in order to achieve competent identity resolution. Identity Resolution, according to the paper, helps to determine if an identity is the same even after it has been described differently. This paper establishes that there are three main Identification attributes which are Personal Identification Attribute, Social Identity Attribute and Social Relationship Attribute. The work was carried out using personal attribute and social attributes to aid in identity resolution. This paper suggests that Identity Attributes and Matching Algorithms are the solutions to developing a frame work of identity resolution. The paper talked about using a set of references and a set of unknown individuals, pairing references with each other and also of incidents in order to get the identity of an individual. The matching algorithms used were unsupervised methods which comprises of Pairwise Comparison, Transitive Closure and Collective Clustering. In the on-going work, synthetic data sets were used for computation and evaluation while during testing and validation, real data sets were used.

**When to conduct Probabilistic Linkage Versus Deterministic Linkage; A Simulation Study:**

Zhu et al., (2015) conducted a study on when to use Deterministic Linkage and when to use probabilistic linkage during data comparison. The simulation study as aimed at understanding the particular characteristics of data that influences the performance of deterministic linkage and probabilistic linkage. Non-unique identifiers were used. To increase linkage patterns and also to increase difficulties, a range of discriminative power was introduced and the sizes of the files, the missing rates and error rates were varied. The performance of both linkage techniques were measured using standard validation methods (such as harmonic mean of sensitivity, Positive Predictive Value (PPV) and f-measure) and computation time. Using PPV, deterministic linkage proved better but validation with sensitivity worked better on probabilistic linkage. The study showed that the rate of missing values and errors was key in choosing linkage methods. From the study, generally, probabilistic linkage was a better linkage technique than deterministic linkage except in cases where the data had very low error rate ($\leq$ 5% error).

**Probabilistic Linkage:**

Winkler (2015) developed a comprehensive study on probabilistic linkage as a technique for data comparison and record linkage. The paper started with an introduction to probabilistic matching algorithms and then went ahead to discuss on the applications of probabilistic matching algorithms. Different algorithms under probabilistic matching algorithm were analyzed; algorithms such as the Fellegi-Sunter algorithm, the Jaro-Winkler algorithm and the Expectation-Maximization algorithm. The paper also discussed about how linkage can be done with training data and how

it can be done without training data. Machine learning model on how to compare and analyze these datasets were also elaborated on. The standard machine learning model for record linkage is the Naïve Bayes Model although Winkler argued that Machine Learning Models like Support Vector Machine and Boosting typically outweigh Naive Bayes Classifier and Logical Regression in performance. String comparators, indexing and blocking as tools for comparisons of records in datasets were explained with examples. Winkler concluded that if extensive 'edit' rules are available from experts or if there is availability of exceptionally clean auxiliary files, then it might be possible to eliminate false matches during the comparison of the datasets. Winkler has suggested such edit rules to be used in general imputation, in his 2011 paper titled 'Cleaning and Using Administrative Lists: Methods and Fast Computational Algorithms for Record Linkage and Modelling/Editing/Imputation and in his 2013 paper titled 'Cleanup and Statistical Analysis of Sets of National Files', in order to reduce the rate of false matches.

**Probabilistic Record Linkage:**

Sayers et al., (2015) discussed probabilistic linkage expressly. The paper was aimed to describe the process of record linkage using a simple exemplar. The method of Deterministic linkage and probabilistic linkage were first of all juxtaposed and compared and then, data structures needed to describe probabilistic linkage were illustrated and described. The paper described the flow of calculating and interpreting matched weights and how to analyze and work on matched weights using Bayes' Theorem. Processes of probability linkage such as pre-data processing, record comparison, indexing, blocking etc. were explained. Match weights were converted into posterior probabilities using Naïve Bayes' Theorem. From the study, it was

concluded that although probabilistic linkage is complex, it gives opportunity for a more robust record linkage than deterministic linkage does. The paper also highlights the various benefits of probabilistic linkage.

**Cross-Device Tracking: Matching Devices and Cookies:**

Diaz-Morales(2015) presented a solution to deal with cross-device identification of users based on semi-supervised machine learning methods that links cookies to an individual using a device. This work was actually a challenge or a competition as to who would get the classifier with the highest $F_{0.5}$ score. The score generally measures the accuracy of the project by calculating the precision and the recall of the project. To create a classifier for this project, a Regularized Boosted Trees algorithm was selected and logistic regression was used for binary classification. The algorithm partitioned the data into sort of clusters and then used the clusters as new ways to further reduce the objective function. A method referred to as bagging was used to improve the stability of machine learning algorithms by reducing variance and avoiding overfitting. The project achieved an accuracy of 0.88% .

**Smart Data Fusion: Probabilistic Record Linkage adapted to Merge Two Trajectories From Different Sources:**

Martinez et al (2018) adapted probabilistic methods in the aviation industry by adapting several techniques to two sources of trajectories namely Radar and GPS. They achieved the aim of not only linking more records than rule-based sorting, but also linking trajectories even when key identifiers have been removed. The paper first did a comparison between probabilistic matching and deterministic matching and then did a proper review on probabilistic matching. A review on metric distances for linkage

was carried out. The paper introduced a brand new model that uses trajectory similarities of flights as a cost function. After the study was over, it was observed that with rule-based linkage, 80% of records were accurately matched, using Hausdorff distance as a cost function, 65% of the records were accurately matched for one day of radar data. This cost function matching was 25% more than matching achieved by deterministic linkage and an almost 50% relative increase.

# CHAPTER 3

## 3.1. METHODOLOGY

The telecommunication industry has a lot of data related to households, individuals and devices. Advertisers pay a premium to ensure they advertise to their target audience. To ensure that content is personalized, it is necessary to accurately predict who is using a device in real time. A probabilistic matching algorithm is developed to determine the profile of an individual based on behavioral analytics. This project is aimed at leveraging probabilistic matching algorithms to create an identity graph that links individuals to their respective devices.

In using probabilistic matching algorithms, machine learning methods are needed in order to successfully carry out a robust linkage of datasets. Two datasets were linked, compared and matched with each other**; the People dataset and the Device dataset.** In probabilistic matching, there is the master file and the file of information; the master file is the People dataset and the file of information is the Device dataset.

The aim of this project is not just to use the personal identity of an individual to identify them, but also to use the social identity and social behaviours in identifying individuals and matching their information from one dataset to another dataset.

The use of hardware for the project was minimal because this was basically a software project. The programming language used for this matching is the python programming language. The environment used to run this language is the jupyterlab that comes installed on the anaconda navigator package.

Due to the fact that no dataset was trained before this work was carried out, the machine learning method used in this work is the **Unsupervised machine learning method**. The algorithm used to carry out this work is the EM Algorithm (Expectation Maximization Algorithm) using the Fellegi-Sunter probabilistic matching model.

### 3.1.1. GENERAL PROBLEMS OF RECORD LINKAGES THAT REQUIRES PROBABILISTIC MATCHING

1. There is no personal identifier on one or both datasets to be matched.
2. There could be the problem of missing data in the dataset(s).
3. Some information such as gender, address, state etc. could be made available for matching.

## 3.2. PROBABILISTIC MATCHING

### 3.2.1. Probabilistic Matching

Unlike deterministic matching, probabilistic matching takes into account a wider range of identifiers, matches the identifiers for the record pairs being compared from the two files that are being matched and computes the similarity weight of these identifiers. The weight calculated will be used to calculate the probability that two records from two different datasets refer to the same entity. While deterministic matching uses programming languages like SQL to be pre-programmed, data can be trained with little or no human intervention when using probabilistic matching algorithms (Dario Martinez).

Two thresholds $T_u$ (Upper threshold) and $T_l$ (Lower threshold) are calculated. If the calculated probability for a record pair exceeds $T_{u,}$ then the record pairs are deemed to be true matches. If the calculated probability for a record pair is below $T_{l,}$ then the record pairs are said to be non-matches. If the calculated probability of a record pair lies between $T_l$ and $T_{u,}$ then the match for the record pair is undecided.

Probability matching incorporates levels of reliability and discriminatory power within specific identifiers (Bigelow et al., 1999).

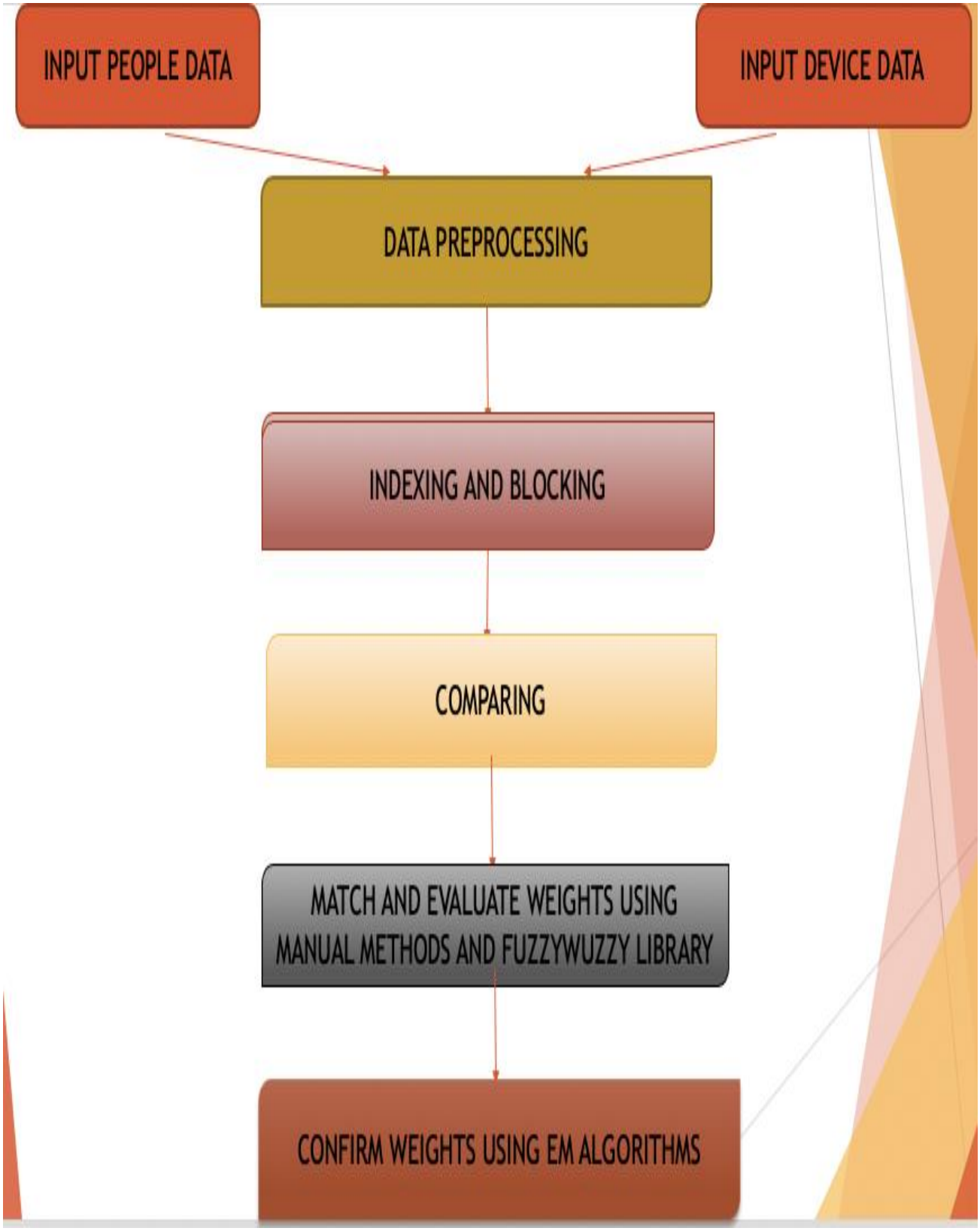


**Figure 1. Probabilistic Matching Process**

### 3.2.2. Performing Probabilistic Matching

1. Link the records through the calculation of probability weights (which shows the likelihood of linkage between records).
2. Adjust data sets for missing data or incomplete data.
3. Probability weights are to be estimated using the observed agreement or disagreement gotten from the agreement pattern (Bigelow et al., 1999).
4. Software can be used in implementing linkage algorithms. In this project, the python software was used for the implementation of the algorithm used which in this case is the k-means algorithm.

### 3.2.3. Probabilistic Matching Algorithm

Many probabilistic matching algorithms assign match or non-match weights to the linkage identifier by using two probabilities **U** and **M.**

**M-probability**

The **m-**probability is the probability that during matching, identifiers from a record pair which is a match will agree. The **m-**probability is also referred to as **Reliability.**

It is usually calculated through manual review or through research on previous works (Bigelow et al., 2015)

It is approximately calculated as:

$M(i) = 1 - error\ rate$

Where **i** is the specific linkage identifier/field.

**U-probability**

The **u-**probability is the probability that during matching, identifiers from a record pair which is **non-matched** will agree. The **u**-probability is also referred to as **Discriminatory power.** It is usually generated from the actual frequency of different values in the datasets.

It is approximately calculated as:

$$U(i) = \frac{1}{number\ of\ values}$$

The linkage weight for a **match** on a given element is calculated as: $\frac{M(i)}{U(i)}$

The linkage weight for a non-match on a given element is calculated as: $\frac{1-M(i)}{1-U(i)}$

Total linkage weight for a record pair is calculated as:

$$(Multiplicative\ sum\ of\ all\ linkage\ weights)\ \times$$
$$(odds\ of\ a\ random\ true\ match\ between\ two\ datasets)$$

If the total linkage weight exceeds the threshold $T_u$, which is the upper threshold, then it can be safely said that the record pair (which comprises of a record from people data and a record from device data) are a match. If the total linkage weight is less than the threshold $T_l$, then the record pair are a non-match. If the total linkage weight is greater than $T_l$ but less than $T_u$, then the match of the record pair is undecided. It will have to be determined by human review (i.e. manually).

Generally, m and u-probabilities are not usually found practically. The m-probability can be easily found if previous work on probabilistic matching has been done on the dataset. One of the easiest ways to calculate m-probability of a field is to count the number of occurrences of a particular entity in the field of interest.

Both datasets used for this work contain 700 rows of data each. Therefore the match space will be 490000 if each row in the people data is matched with each row in the device data. Of the 490000 compared records, there will be at most 700 records that match. Now if the field of interest during linkage is 'l_name' and there are say, 10 Calebs and 20 Smiths

The m-probability of Caleb will be $\frac{\mathbf{10}}{\mathbf{700}} = \mathbf{0.01428}$

The m-probability of Smith will be $\frac{\mathbf{20}}{\mathbf{700}} = \mathbf{0.02857}$

So, the remaining 489,300 are non-matches. From this non-match, the u-probability can be calculated. The 'Caleb' name will have 100 comparisons at most when both files are linked and out of the 100, only 10 are matches, while 'Smith' will have 400 comparisons and only 20 are matches.

The u-probability of Caleb and Smith will be $\frac{\mathbf{90}}{\mathbf{490000}} = \mathbf{0.0001836}$ and $\frac{\mathbf{380}}{\mathbf{490000}} = \mathbf{0.0007755}$ respectively.

Therefore, from the calculations above, the likelihood ratios for agreement for Caleb and smith will be: $\frac{\mathbf{0.01428}}{\mathbf{0.0001836}} = \mathbf{77.77}\ \boldsymbol{and}\ \frac{\mathbf{0.02857}}{\mathbf{0.0007755}} = \mathbf{36.84}$ respectively.

Because Smith is a more common name than Caleb is, it has a lesser discriminatory power than Caleb does (Sayers et al., 2015)

### 3.2.4. Mathematical Implications Of Probabilistic Matching

There are two datasets to be matched; the people dataset $(P_d)$and the device dataset $(D_d)$. Let the records (rows) of the people dataset and the device dataset be denoted

as $\alpha(p)\ and\ \beta(d)$ respectively. Let the linkage fields represented in the files (datasets) be denoted as K.

The set of records that represent similar entities can be defined as:

$$M = \{(p, d): p = d, p \in P_d, d \in D_d\}$$

The set of records that represent different entities can be defined as:

$$U = \{(p, d): p \neq d, p \in P_d, d \in D_d\}$$

In probabilistic matching, there is a vector $\gamma$ known as the agreement pattern of linkage fields. For example, take two records of individuals represented in the people data.

| F_name | L_name | Birth_month | Birth_year | Sex | City | State |
|---|---|---|---|---|---|---|
| Omolade | Odedina | October | 1994 | F | Gwarinpa | Abuja |
| Omolade | Temitope | October | 1993 | F | Galadimawa | Abuja |

**Table 1. Sample records of Individuals**

Generally, the agreement pattern ($\gamma$) will be written as [1,0,1,0,1,0,1]. The reason being that both records of indivuals agree on the F_name, Birth_month, Sex and State which are the first, third, fifth and seventh column and disagree on the L_name, Birth_year and City which are the second, fourth and sixth column. When the fields agree, the number 1 is assigned to their agreement and the number 0 is assigned otherwise.

The agreement pattern $\gamma$ that contains the coded agreement and disagreement of the fields between two records can be defined as:

$$\gamma[\alpha(p), \beta(d)] = \{\gamma^1[\alpha(p), \beta(d)], \ldots, \gamma^k[\ \alpha(p), \beta(d)]\}$$

Therefore, the conditional probabilities of observing a specific vector $\gamma$ given $(p, d) \in M \; and \; (p, d) \in U$ are defined:

$$m(\gamma) = P\{\gamma[\alpha(p), \beta(d)|(p, d) \in M]\} = \sum_{(p,d)\in M} P\{\gamma[\alpha(p), \beta(d)]\}.P[(p, d)|M]$$

And

$$u(\gamma) = P\{\gamma[\alpha(p), \beta(d)|(p, d) \in U]\} = \sum_{(p,d)\in U]} P\{\gamma[\alpha(p), \beta(d)]\}.P[(p, d)|U]$$

### 3.2.5. The Fellegi-Sunter Model:

Fellegi and Sunter (in their 1969 paper, A Theory for Record Linkage) provided a formal mathematical model for ideas that were introduced by Howard Newcombe. In this method, two files, say files A and B are to be matched to each other where the cross match A × B should give a set of true matches represented as M and a set of true non-matches represented as U. Records in file A are to be compared with records in file B such that if these two records refer to the same entity, they are agreed to be a match but if they refer to different entities, they are said to be non-matches. Fields names are combined to give what is referred to as an agreement pattern denoted by the symbol ($\gamma$) (Sayers et al., 2015)

The Fellegi-Sunter Method provides an optimal method in which the set of possible links, during the pairing of records from two datasets, can be minimized.

### 3.2.6. Processes In Probabilistic Matching

#### I. Data collection:

It is a process of gathering information and measuring information on targeted segments in a system. It is regarded as one of the main component of research

in any field. The methods of data collection vary with respect to the research being carried out and even the researcher. Data can be collected through the use of survey, questionnaires, google forms etc.

Due to the nature of this project, two different datasets were collected so that records from one dataset can be compared and /or matched with records from other datasets. The first dataset is referred to as the '**people data'** while the second dataset is referred to as the '**device data'.**

The people data contains the information of the entity (individual, in this case) to be identified. The people dataset contains 700 rows and 9 columns (f_name, l_name, age, gender, average_daily_int_gb, social_interests, state, serial_no_dev, mac_add).

The device data contains the information of the devices (the project was scaled to mobile phones because these mobile phones are literally always with individuals, making it easier for them to see advertisement when they pop-up). The device dataset comprises of 700 rows and 6 columns. Information such as serial_no_dev, mac_add, ad_click_rate, average_daily_int_gb, ip_add and social_interests.

**People dataset:**

**F_name**: Represents the first name of the individual

**L_name:** Represents the last name of the individual

**Average_daily_int_gb:** Represents the average daily internet usage of the individual in gigabytes.

**Social_interests:** Represents the behavior of the individual on the internet and the social likes of such individuals. Known also as the behavioural pattern.

**Serial_no_dev:** Represents the serial number of the device owned by the individual

**Mac_add:** Represents the MAC address of the device

```
In [9]: people_data
```

Out[9]:

| | f_name | l_name | age | gender | monthly_income | average_daily_internet_usuage | social_interests | state | serial_no_dev | mac_add |
|---|---|---|---|---|---|---|---|---|---|---|
| 0 | Caesar | Marunchak | 35 | Male | 61833.90 | 256.09 | Crime technology romance | Abuja | 999711451-5 | 1B-FD-10-4F-8C-40 |
| 1 | Melvyn | Borland | 31 | Male | 68441.85 | 193.77 | Film making cameras shoes | Ogun | 505470266-0 | 0C-6A-14-73-A4-B7 |
| 2 | Clementia | Wisden | 26 | Male | 59785.94 | 236.50 | Celebrities cybersecurity engineering | Lagos | 159922092-X | E4-92-7A-09-09-DF |
| 3 | Paco | Geraud | 29 | Male | 54806.18 | 245.89 | Buildig technology constructions | Abuja | 553505766-5 | 3D-2E-9C-D7-65-DC |
| 4 | Umeko | Prettejohns | 35 | Male | 73889.99 | 225.58 | Software TEDex crime | Lagos | 847021967-7 | 4D-80-F2-D8-22-BB |
| 5 | Stevy | Fried | 23 | Male | 59761.56 | 226.74 | Metro cars romance people | Lagos | 670081497-9 | F8-8E-2E-8B-3B-F6 |
| 6 | Cobby | Fleeming | 33 | Male | 53852.85 | 208.36 | Business website development | Ogun | 198918308-5 | 6F-15-47-52-64-F7 |
| 7 | Rich | Truter | 48 | Male | 24593.33 | 131.76 | Data science machine learning artificial intel... | Ogun | 081211569-4 | 97-BA-5E-BC-97-22 |
| 8 | Artair | Coldtart | 30 | Male | 68862.00 | 221.51 | Public speaking leadership writing | Lagos | 878471562-6 | B0-C6-EB-9D-0C-7D |
| 9 | Lacie | Hemeret | 20 | Male | 55642.32 | 183.82 | Graphic design pr UI UX | Lagos | 666019915-2 | 85-0D-6C-E4-A8-6E |

**Figure 2. An Overview of People Dataset**

**Device dataset:**

**Ad_click_rate:** The rate at which an advert is clicked on from the device. Allows us to know how often an advert can be sent to a device owned by an individual.

**Ip_add:** IP address of the network used by the individual.

In [10]:
```
device_data
```

Out[10]:

| | serial_no_dev | mac-add | ad_click_rate | ip_add | social_interests |
|---|---|---|---|---|---|
| 0 | 999711451-5 | 1B-FD-10-4F-8C-40 | 77.17 | 185.78.224.130 | Crime technology romance |
| 1 | 505470266-0 | 0C-6A-14-73-A4-B7 | 87.79 | 130.170.252.154 | Film making cameras shoes |
| 2 | 159922092-X | E4-92-7A-09-09-DF | 15.16 | 213.233.109.17 | Celebrities cybersecurity engineering |
| 3 | 553505766-5 | 3D-2E-9C-D7-65-DC | 99.39 | 237.174.86.15 | Buildig technology constructions |
| 4 | 847021967-7 | 4D-80-F2-D8-22-BB | 53.64 | 87.85.128.173 | Software TEDex crime |
| 5 | 670081497-9 | F8-8E-2E-8B-3B-F6 | 30.60 | 20.232.131.220 | Metro cars romance people |
| 6 | 198918308-5 | 6F-15-47-52-64-F7 | 63.32 | 207.125.195.13 | Business website development |
| 7 | 081211569-4 | 97-BA-5E-BC-97-22 | 69.97 | 81.143.8.169 | Data science machine learning artificial intel... |
| 8 | 878471562-6 | B0-C6-EB-9D-0C-7D | 63.59 | 43.231.41.54 | Public speaking leadership writing |
| 9 | 666019915-2 | 85-0D-6C-E4-A8-6E | 34.87 | 199.231.154.49 | Graphic design pr UI UX |
| 10 | 235769345-2 | 24-BE-56-68-91-63 | 64.73 | 1.139.82.120 | Centralized neutral neural-net |
| 11 | 299330461-5 | BC-78-6C-AA-AD-BF | 26.81 | 102.36.18.47 | Team-oriented grid-enabled Local Area Network |
| 12 | 564491848-3 | D6-76-DB-28-79-61 | 9.42 | 247.14.189.103 | Centralized content-based focus group |
| 13 | 647593171-3 | 71-D1-95-4E-11-F4 | 10.05 | 186.5.150.3 | Synergistic fresh-thinking array |
| 14 | 821331822-6 | 88-9D-11-04-8E-E8 | 76.67 | 118.60.156.239 | Grass-roots coherent extranet |
| 15 | 101544698-1 | 1A-BF-81-C2-9A-A6 | 14.19 | 75.74.71.15 | Persistent demand-driven interface |
| 16 | 174759040-1 | 5E-A9-7A-42-AE-E7 | 23.77 | 188.71.102.58 | Customizable multi-tasking website |
| 17 | 424617561-7 | 94-D2-39-6B-05-81 | 42.17 | 143.111.116.243 | Intuitive dynamic attitude |
| 18 | 240176280-9 | CA-F4-D0-F6-3D-D8 | 2.81 | 163.110.241.82 | Grass-roots solution-oriented conglomeration |
| 19 | 951639059-5 | 32-F9-12-66-2D-0A | 95.90 | 92.223.157.255 | Advanced 24/7 productivity |
| 20 | 058960061-3 | CF-D7-F7-28-E2-C5 | 45.50 | 13.180.194.55 | Object-based reciprocal knowledgebase |

**Figure 3. An Overview of the Device Dataset**

## II. Data Preprocessing:

This process is very applicable in data mining and machine learning projects. Generally, data is usually inconsistent, prone to error or even worse, consist of missing values. Data preprocessing does the work of cleaning these datasets Data preprocessing is simply the conversion of a data into a way that is comprehensible for the user of the dataset. The different processes involved in data preprocessing can include data cleaning, normalization, feature extraction etc.

In this work, there was not much of cleansing to do because the dataset was already cleaned from its source.

III. **Data Linkage:** The data linkage process was divided into four parts.

i. **Data Deduplication:**

First, in probabilistic matching, it is always necessary to deduplicate (i.e. remove any duplicate records in your dataset). This ensures linkage is more accurate.

ii. **Indexing and blocking:**

Indexing is done when each record in people dataset is paired with each record in the device dataset. There are two main types of indexing during probabilistic matching. There is the full indexing and there is the partial indexing. The full indexing is the general type of indexing but it is usually avoided especially on large datasets. For example, if there are two different files of 1,000,000 rows each, performing full index pairing will give 1,000,000,000,000 record pairs which might take a very long time to run. Therefore, when matchings are being carried out on large datasets, full indexing is usually avoided. The partial indexing on the other hand is what is generally referred to as 'blocking'. When it has to deal with indexing, blocking is generally referred to as the standard indexing method. When performing blocking operations on datasets, a blocking key is taken into consideration. The blocking key can comprise of one or more common fields in both datasets. The blocking key is chosen such that when the records are paired, any of the records which do not have anything similar when those blocking keys are applied are considered a non-match. With this, the comparison space is minimized and run time is also optimized when running the code.

```
In [25]: indexer = recordlinkage.Index()
         indexer.full()
         pairs = indexer.index(people_data, device_data)

         print(len(people_data))
         print(len(device_data))
         print(len(pairs))
```

```
WARNING:recordlinkage:indexing - performance warning - A full index can result in large number of record pairs.
700
700
490000
```

**Figure 4. An Overview of Full Indexing Pairing**

```
In [26]: #indexing
         indexer = recordlinkage.Index()
         indexer.block("mac_add")
         indexer.block("serial_no_dev")
         indexer.block("social_interests")
         pairs = indexer.index(people_data, device_data)

         print(len(people_data))
         print(len(device_data))
         print(len(pairs))
```

```
700
700
557
```

**Figure 5. An Overview of Partial Indexing Pairing**

The warning can be seen in Figure 4 above, that a full index could result in a large number of record pairs and which could affect the performance of the program. From the diagram above, it can be seen that when a full index was

performed on the data, there were a total of 490,000 pairs because the indexing was done record by record where the each dataset had 700 rows but when a blocking was done using the ‘mac_add’, ‘serial_no_dev’ and ‘social_interests’ fields, the total linkage was reduced drastically to 557 instead of 490,000. If a matching record pair is not included in the indexing, it cannot be matched anymore. So care has to be taken in order to ensure it is done efficiently.

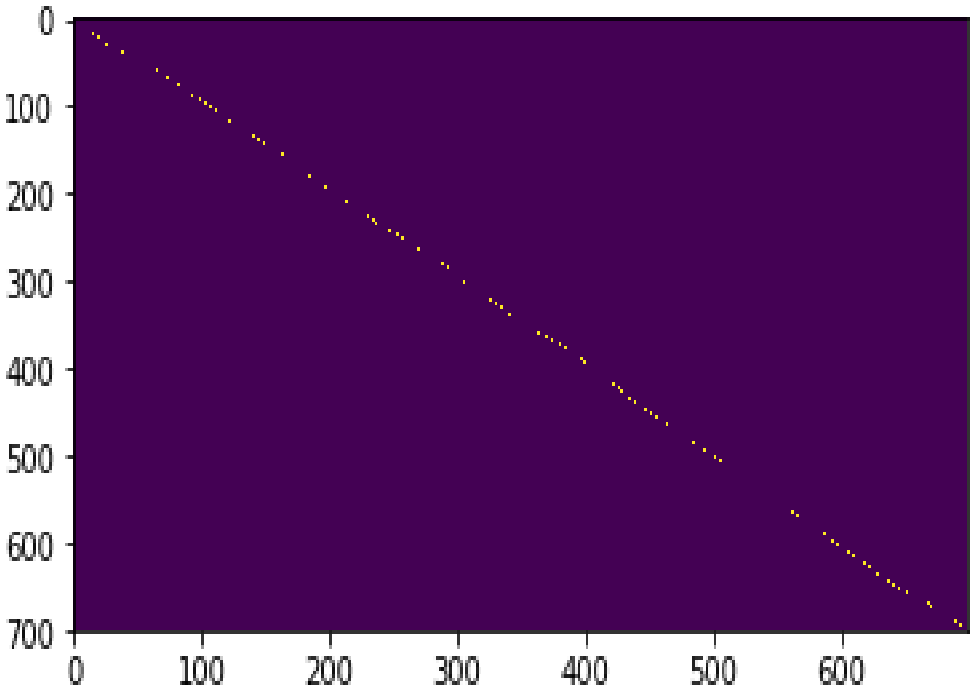


**Figure 6. Visualization of the Blocking Process**

If the image of the blockings is viewed properly, it is observed that there are small dotted clusters that form a line that is almost straight. This is due to the fact that almost all the records in both datasets do not have other records that belong to the same cluster with them. For datasets that have records that belong to the same cluster after blocking, the clusters are usually larger and haphazard in shape.

**iii. Comparing:**

Generally, when performing probabilistic linkage of two records manually, the comparison between the record pairs are usually done attributes by attributes. It compares the records on all the fields both records in the datasets have in common. The common fields to both datasets are; average_daily_int_gb, mac_add, social_interests and serial_no_dev. The comparison of record pair is usually done using what is known as the **similarity function**.

Different methods are used when comparing two fields. For example, when comparing strings, methods like edit distance, SOUNDEX, Euclidean distance etc are used to compare the dissimilarity between string characters. The major comparison method used for numbers is the Jaro-Winkler method. The two methods used in the python library that was used in this work (record linkage) includes are the Edit distance and the Jaro-Winkler method.

a. **The Edit distance:** it is simply the amount of operations (insertions, deletions, substitute etc.) that are needed to be carried out on a string in order for it to be converted into another string.
b. **Jaro-Winkler method:** Simply compares common characters in a string even if the characters are transposed (Jaro); Winkler later on worked on new methods to upweight similarities at the beginning of strings.

   The Jaro method is given as:

$$Jaro(S_1, S_2) = w_1 \frac{C}{L_1} + w_2 \frac{C}{L_2} + w_t \frac{C - \tau}{C}$$

Where: L1 = length of string 1

L2 = length of string 2

C = number of characters both strings have in common

$\tau$ = number of transpositions

$w_1, w_2$ and $w_t$ = weights

The Jaro-Winkler method is given as:

$$Jaro - Winkler(S_1, S_2) =$$

$$Jaro(S_1, S_2) + 0.1 \times i \times (1 - Jaro(S_1, S_2)$$

Where: i = min(4, number of initial characters that match)

**c. Jaccard distance method:** It is used in cases where sets of strings are being compared to each other and the sequence of the sets do not matter. It takes into consideration the content of the string rather than the sequence of the string.

Say there are two set of strings to be compared, A and B, the Jaccard distance between the string set is given as:

$$Jaccard\ distance = \frac{L(A \cap B)}{L(A \cup B)}$$

Where: $L(A \cap B)$ = length of A∩B

$L(A \cup B)$ = length of $A \cup B$

**iv. Classification:**

The classification Algorithm used for this project is the EM Algorithm (sometimes known as the Expectation/Conditional Maximization Algorithm). It used the linkage rules to divide the search space

$(People\ data\ \times\ device\ data)$ into a set of designated matches, possible matches/non-matches, designated non-matches (Winkler, 2000)

Popularly known as the EM Algorithm. It is an Iterative algorithm that we use in order to get the m and u – probabilities to be used in the probabilistic matching.

The EM Algorithm helps us to get the weights and probabilities for which we can decide if records indeed match.

**The simplest form of the EM Algorithm can be written as:**

1) Replace missing values with estimated values

2) Estimate parameters

3) Re-estimate values for the missing data with the new parameters

4) Repeat until convergence

Each step above will be discussed in details in the next section.

# CHAPTER 4

This project was evaluated, tested and validated after implementation, in three parts. The people dataset and device dataset were linked, compared and matched to form a large set of matched and unmatched data.

The first part of the test was the calculation of the Positive Predictive Value (PPV) and the match score of the matched records to check the match score of the matched datasets. Three methods were used to carry out this section and these three methods were compared to one another to indeed confirm if the PPV value was estimated correctly and if indeed the match score was what was estimated.

The second section to be tested are the string comparators used for the project. Three string comparators ‘were used for comparing string and other objects in the data sets namely The Jaro-winker distance comparator, the Edit distance comparator and the Jaccard comparator. Each of these string comparators were run on the matched datasets and the unmatched datasets and then the visible results of the matching were plotted in order to test for which string comparator performed the most.

The third section of the test is to show an over view of the matched records as a dataset on its own which if observed, will show that indeed the record pairs matched belonged to the same entity and signifies a high level of accuracy.

The match score works such that, if the match score of a paired record is below it, the record pair is considered a non-match but for any score above it, the record pair is considered a match.

## 4.1. MATCH SCORE AND POSITIVE PREDICTIVE VALUE (PPV)

### 4.1.1. Manual Calculation of Match Score and PPV

Cut-off threshold is defined generally as the difference between the desired weight and the estimated weight.

**Estimated Weight**

With the People data and the Device data, the estimated weight for each pair of records is equal to the $\log_2$ of the chances of picking true matches by coincidence.

It is represented as:

$$Estimated\ weight = \log_2 \left( {}^{E}/_{(A \times B) - E} \right)$$

Where: E = Number of expected matches = 700

A = Number of records in people data

B = Number of records in device data

Therefore,

$$Estimated\ weight = \log_2 \left( {}^{700}/_{((490000 - 700))} \right)$$

Therefore, Estimated Weight = -9.449

**Desired Weight**

To calculate this, a desired PPV is chosen, for this project, the PPV chosen is 0.95. The desired weight for each pair of records is the $\log_2$ of the chances associated with the PPV.

It is represented as:

$$Desired\ weight = \log_2 \left( {}^{P}/_{(1 - P)} \right)$$

Where P = Positive Predictive Value

Therefore,

$$Desired\ weight = \log_2 \left( {}^{0.95}/_{(1 - 0.95)} \right)$$

$$Desired\ weight = \log_2 \left( {}^{0.95}/_{0.05} \right)$$

Therefore, Desired weight = 4.25

**Cut-off threshold**

The cut-off threshold, also known as the match score can now be gotten from the Estimated weight and the desired weight.

It is represented as:

$$cut - off\ threshold = Desired\ weight - Estimated\ weight$$

$$cut - off\ threshold = 4.25 - (-9.449)$$

Therefore, cut-off threshold (match score) = 13.699

Using the manual method; **PPV value = 0.95**

**Match score = 13.699**

I went ahead to confirm this scores and values using the fuzzywuzzy method and the EM Algorithms.

### 4.1.2. Using Fuzzywuzzy Library for PPV and Match Score

A python library, Fuzzywuzzy, was used to also confirm the PPV and Match Score. After the matching was was done, different match scores were used to measure and determine the PPV of the matching.

```
(len(data_merge[data_merge.apply(get_ratio, axis=1) >13])) / len(data_merge)
```

```
0.9714924538848518
```

**Figure 7. Matching with a match score of 13**

```
(len(data_merge[data_merge.apply(get_ratio, axis=1) >14])) / len(data_merge)
```

```
0.9575181665735047
```

**Figure 8. Matching with a match score of 14**

```
(len(data_merge[data_merge.apply(get_ratio, axis=1) >15])) / len(data_merge)
```

```
0.9496925656791504
```

**Figure 9. Matching with a match score of 15**

| MATCH SCORE | POSITIVE PREDICTIVE VALUE |
|---|---|
| 13 | 0.9715 |
| 14 | 0.9575 |
| 15 | 0.9497 |

**Table 2. Match Score with corresponding Positive Predictive Values**

### 4.1.3. Comparing Manual Matching, Fuzzywuzzy and EM Algorithms

It is observed, that while using the manual method to determine the match score, the PPV were chosen by the researcher and while using the Fuzzywuzzy method, the

match score were picked by the researcher to determine the PPV but the EM Algorithm was able to determine both the PPV and the match score.

From the findings above, it can be said that the match score with which the records are to be matched is between **13** and **15** while the PPV is around **0.95 (95%).**

## 4.2. STRING COMPARATORS

Three string comparators – Edit distance, Jaro-Winkler and Jaccard– were used for comparison and the performances of each were compared with others after the data has been matched. The figure below shows the performance of the three string comparators on the matched datasets.

From the diagram, it is easily observed that the Jaro-winkler method gave the highest performance for high match scores. That is why the Jaro-Winkler method is a much preferred string comparator for record linkage. The comparator with the second best performance is the Jaccard comparator followed by the edit distance.

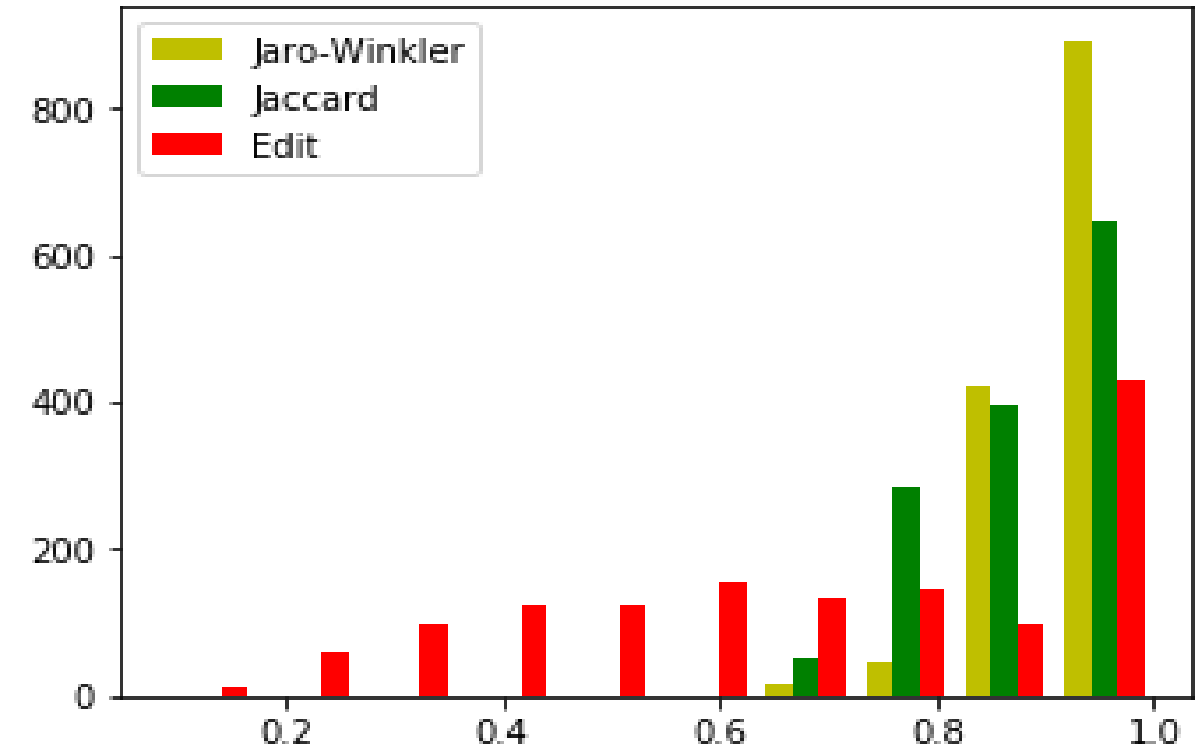


**Figure 10. Performance of the Three String Comparators Used**

### 4.3. THE MATCHED DATASETS

After careful implementation, records were linked, compared and matched. The matched records formed another dataset, the unmatched records formed another dataset. The figure below gives an overview of the matched data.

| erage_daily_int_gb | social_interests_x | state | serial_no_dev_x | mac_add_x | serial_no_dev_y | mac_add_y | ad_click_rate | ip_add | social_interests_y |
|---|---|---|---|---|---|---|---|---|---|
| 2.65 | Crime technology romance | Abuja | 999711451-5 | 1B-FD-10-4F-8C-40 | 999711451-5 | 1B-FD-10-4F-8C-40 | 77.17 | 185.78.224.130 | Crime technology romance |
| 2.08 | Film making cameras shoes | Ogun | 505470266-0 | 0C-6A-14-73-A4-B7 | 505470266-0 | 0C-6A-14-73-A4-B7 | 87.79 | 130.170.252.154 | Film making cameras shoes |
| 2.08 | Fully-configurable neutral open system | Taraba | 871723033-0 | 0C-B2-13-B2-FB-DE | 871723033-0 | 0C-B2-13-B2-FB-DE | 68.29 | 113.45.188.2 | Fully-configurable neutral open system |
| 2.08 | Public-key non-volatile implementation | Abuja | 632586842-0 | FB-8C-AF-95-8C-07 | 632586842-0 | FB-8C-AF-95-8C-07 | 14.59 | 133.149.164.81 | Public-key non-volatile implementation |
| 1.05 | Celebrities cybersecurity engineering | Lagos | 159922092-X | E4-92-7A-09-09-DF | 159922092-X | E4-92-7A-09-09-DF | 15.16 | 213.233.109.17 | Celebrities cybersecurity engineering |
| 1.58 | Buildig tchology constructions | Abuja | 553505766-5 | 3D-2E-9C-D7-65-DC | 553505766-5 | 3D-2E-9C-D7-65-DC | 99.39 | 237.174.86.15 | Buildig technology constructions |
| 3.65 | Software TEDex crime | Lagos | 847021967-7 | 4D-80-F2-D8-22-BB | 847021967-7 | 4D-80-F2-D8-22-BB | 53.64 | 87.85.128.173 | Software TEDex crime |
| 3.10 | Metro cars romance people | Lagos | 670081497-9 | F8-8E-2E-8B-3B-F6 | 670081497-9 | F8-8E-2E-8B-3B-F6 | 30.60 | 20.232.131.220 | Metro cars romance people |
| 4.95 | Business website development | Ogun | 198918308-5 | 6F-15-47-52-64-F7 | 198918308-5 | 6F-15-47-52-64-F7 | 63.32 | 207.125.195.13 | Business website development |
| 3.59 | Data science machine learning artificial intel... | Ogun | 081211569-4 | 97-BA-5E-BC-97-22 | 081211569-4 | 97-BA-5E-BC-97-22 | 69.97 | 81.143.8.169 | Data science machine learning artificial intel... |

**Figure 11. An overview of the matched data**

# CHAPTER 5

## 5.1. CONCLUSION

It can be concluded that EM Algorithm is one of the most efficient algorithm for probabilistic matching. The EM algorithm is preferred in that it uses an unsupervised technique of machine learning, where there is no need for any training and test data. This makes the EM Algorithm easier to work with. The results came out and discussions were inferred from the results obtained.

The manual matching and Fuzzywuzzy were used to obtain match score and PPV, while the EM Algorithm was used to confirm the match score and the PPV. The match score for all record pairs was between **13** and **15**, while the PPV was about **0.95 (95%)**. After this matching, devices can be linked to individuals, making it easy, therefore, for advertisers to recognize their target audience and then advertise the right products to the right people, thereby, increasing the chances of these individuals purchasing products being advertised.

## 5.2. CHALLENGES

**i.** Since probabilistic matching has not been carried out extensively on datasets in the telecommunication industry, the telecommunication data was difficult to come across. Hence, a small dataset had to be used to carry out this project work.

**ii.** The project was supposed to be done on a big data platform but due to the small size of the project, I had to use the python programming language for the project.

## 5.3. RECOMMENDATION

It was observed that Nigerians do not register their phones, thereby making it difficult for researchers to obtain details of devices of individuals from the retailers. I recommend that the government ensure that individuals register their phones as soon as they buy their devices.

## 5.4. CONTRIBUTIONS

Behavioural Analytics was taken into consideration and not just personal and physical attributes. Also, probabilistic matching has not been extensively used in the telecommunication industry and so this work attempts to use probabilistic matching to link individuals to their respective devices.

## 5.5. FUTURE WORKS

Future works can be done using fingerprints patterns to match since it is plausible for behaviours of individuals to change overtime. Hence, the need for the use of fingerprints (which are often not likely to change) for matching individuals with respective devices.

## 5.6. APPLICATIONS OF PROBABILISTIC MATCHING

1. **Medical Practice and Research:**

One of the most common sectors in which probabilistic matching is usually applied is the medical sector. It is an important process when the need for examining the health of the public is required. Data sources can be used to eliminate duplicate records, notice missing people etc.

2. **Historical Research:**

Another sector which popularly uses probabilistic matching is the Historical sector. Since a lot of datasets were recorded long before the invention of identification numbers for individuals (in the case of Nigerians, Bank Verification Number (BVN)), the linking of datasets for longitudinal study is of necessity.

3. **Data Warehousing and Economic Intelligence:**

Data warehousing seeks to combine different datasets from different sources into a lone model which can then be sent into an intelligent system for reporting and analysis.

# APPENDIX

```
%matplotlib inline
import matplotlib.pyplot as plt
import pandas as pd
import re

people_data = pd.read_excel("/Users/FME/Desktop/Everything
project/codes/projectdata/people_data.xlsx")
device_data = pd.read_excel("/Users/FME/Desktop/Everything
project/codes/projectdata/numbers.xlsx")
print("\n", people_data.info())
print("\n",device_data.info())
print("\n",people_data.describe())
print("\n",device_data.describe())


npeople_d = people_data[['mac_add', 'social_interests', 'serial_no_dev',
'average_daily_int_gb']].drop_duplicates()
ndevice_d = device_data[['mac_add', 'social_interests', 'serial_no_dev',
'average_daily_int_gb']].drop_duplicates()
print(len(npeople_d), len(ndevice_d))


from fuzzywuzzy import fuzz
```

```
import pandas as pd
people_data = pd.read_excel("/Users/FME/Desktop/Everything
project/codes/projectdata/people_data.xlsx")
device_data = pd.read_excel("/Users/FME/Desktop/Everything
project/codes/projectdata/numbers.xlsx")
device_data
data_merge = people_data.merge(device_data, on='average_daily_int_gb')
data_merge.head()
def get_ratio(row):
    name = row['social_interests_x']
    name1 = row['social_interests_y']
    name4 = row['mac_add_x']
    name5 = row['mac_add_y']
    return fuzz.token_set_ratio(name, name1,name4,name5)
data_merge[data_merge.apply(get_ratio, axis=1) > 80]

npeople_d['nmac_add'] = npeople_d.mac_add.apply(lambda
x:x.lower().translate({None:"'.,-"}) if pd.notnull(x) else '')
ndevice_d['nmac_add'] = ndevice_d.mac_add.apply(lambda
x:x.lower().translate({None:"'.,-"}) if pd.notnull(x) else '')
df_merge= ndevice_d.merge(npeople_d, on='nmac_add')

import jellyfish
import nltk
```

```
import itertools
def check_triangle(f):
    t0 = u'abc'
    p = [''.join(p) for p in itertools.permutations(t0)]

    for t1 in p:
        for t2 in p:
            if(f(t0, t2) > (f(t0, t1) + f(t1,t2))):
                print("d({t0}-{t2})={d02}  d({t0}-{t1}) + d({t1}-
{t2})={d01_12}".format(t0=t0,t1=t1,t2=t2,d02=f(t0, t2), d01_12=f(t0, t1) + f(t1,t2)))
from __future__ import division

def jaccard_similarity(a, b):
    x = set(a)
    y = set(b)
    return len(x & y) / len(x | y)



print(jaccard_similarity('omolade', 'temitope'))

check_triangle(lambda x,y : 1 - jellyfish.jaro_winkler(x, y))
def jaccard_distance(a, b): return 1 - jaccard_similarity(a, b)
print(jaccard_distance('omolade', 'molode'))
print(jaccard_distance('abcdef', 'cbfaed'))
check_triangle(jaccard_distance)
```

# REFERENCES


1. Li. J and Wang. A. G. 2015. A framework of Identity Resolution: evaluating identity attributes and matching algorithms. Security Informatics; a SpringerOpen journal. DOI 10.1186/s13388-015-0021-0.
2. Winkler W. E. 2015. Probabilistic Linkage. Methodological Development in Data Linkage.
3. Oracle Corporation. June 2008. An Introduction to Oracle Identity Management. An Oracle White Paper. June 2008.
4. Diaz-Morales. R. 2015. Cross-Device Tracking: Matching Devices and Cookies. IEEE 15th International Conference on Data Mining Workshops. Page 30. 33428.
5. Bigelow. W, Karlson. T and Beutel. P. 1999. Using Probabilistic Linkage to Merge Multiple Data Sources for Monitoring Population Health. Centre for Health Systems Research Analysis.
6. Li. J, Wang. G and Chen. H. 2006. Identity Matching Based on Probabilistic Relational Models. Proceedings of the Twelfth Americas Conference on Information System.
7. Zhu. Y, Matsuyama. Y, Ohashi. Y and Setoguchi. S. 2015. When to conduct Probabilistic Linkage Versus Deterministic Linkage; A Simulation Study. Journal of Biomedical Informatics 56. 80-86.
8. Wang. C, Gavrilova. M, Luo. Y and Rokne. J. An Efficient Algorithm for Fingerprint Matching. IEEE.
9. Sayers. A, Ben-Shlomo. Y, Blom. A. W and Steele F. 2015. Probabilistic Record Linkage. International Journal of Epidemiology. PP 1 – 11
10. Li. J, Wang. A. G and Chen. H. 2011. Identity Matching Using Personal and Social Identity Features. Inf Syst Front. PP 101-113. DOI 10.1007/s10796-010-9270-0

**11.** Camps. O. I, Shapiro. L. G and Haralick R. M. 1994. A Probabilistic Matching Algorithm for Computer Vision. Annals of Mathematics and Artificial Intelligence 10. PP 85-124.

**12.** Winkler E. W. Using the EM Algorithm for Weight Comparison in the Fellegi Sunter Method of Record Linkage. 2000. Statistical Research Report Series. No. RR2000/05

**13.** Sayers. A, Ben_Shlomo. Y, Blom. A. W, Steele. F. 2015. Probabilistic Record Linkage. International Journal of Epidermiology. PP 1-11. Doi: 10.1083/ije/dyv322.

**14.** https://recordlinkage.readthedocs.io/en/latest/notebooks/link_two_dataframes.html

**15.** https://www.ilantus.com/history-identity-access-management/