# Abstract

Online dating platforms use a wide variety of ways to recommend matches for an existent person. The dating industry was valued at 3.08 billion GBP in 2019 and is growing rapidly (eharmony editorial team 2015). Popular dating sites are currently focusing on gamification rather than upgrading their offerings, which means they tend to have biases that favor a small group of people over others. Popular dating sites are currently focusing on gamification rather than upgrading their offerings, which means they are biased toward a small set of people. However, personality is frequently underutilized. A neural network with an attention mechanism for bidirectional long short-term memory (LSTM) was created to predict a user's personality in order to see if personality might be utilized to connect users. The matching algorithm was based on how similar two users' personalities were. Users can communicate with a chatbot named Lily and manage their matches through native Android app. Lily collects data for personality prediction by asking standard dating questions throughout the day. Some users were afraid to use a chatbot, but others like the app's simplicity and the ability to have personal conversations with Lily.

Contents

[Abstract i](#_Toc90125124)

[Chapter 1 1](#_Toc90125125)

[1. Introduction 1](#_Toc90125126)

[1.1. Relevance of the research problem 1](#_Toc90125127)

[1.2. Motivation 2](#_Toc90125128)

[1.3. Aim 2](#_Toc90125129)

[1.4. Problem Statement 2](#_Toc90125130)

[1.5. Research Question 2](#_Toc90125131)

[1.6. Research Objective 3](#_Toc90125132)

[1.7. Proposed Rich Picture 3](#_Toc90125133)

[1.8. Challenges and Limitations 4](#_Toc90125134)

[Chapter 2 5](#_Toc90125135)

[1. Literature review 5](#_Toc90125136)

[2.1. Introduction 5](#_Toc90125137)

[2.2. Chatbots 6](#_Toc90125138)

[2.2.1. Rule-Based Chatbots 6](#_Toc90125139)

[2.2.2. Generative Chatbots 7](#_Toc90125140)

[2.2.3. Retrieval-Based Chatbot 7](#_Toc90125141)

[2.3. Dating Algorithms 8](#_Toc90125142)

[2.3.1. Profile Matching 8](#_Toc90125143)

[2.3.2. Collaborative Filtering 8](#_Toc90125144)

[2.2.3. Elo Score 9](#_Toc90125145)

[2.4. Predicting Personality 10](#_Toc90125146)

[2.4.1. Big Five Personality Model 10](#_Toc90125147)

[2.4.2. Essays Dataset 11](#_Toc90125148)

[2.4.3. Linguistic Inquiry and Word Count 12](#_Toc90125149)

[2.4.4. Word Embeddings 12](#_Toc90125150)

[2.4.5. Neural Networks 13](#_Toc90125151)

[2.4.6. Convolutional Neural Networks 15](#_Toc90125152)

[2.4.7. Long Short-Term Memory Neural Networks 16](#_Toc90125153)

[2.4.8. BERT and RoBERTa 18](#_Toc90125154)

[Chapter 3 19](#_Toc90125155)

[2. Methodology 19](#_Toc90125156)

[3.1. Introduction 19](#_Toc90125157)

[3.2. Methodological approach 19](#_Toc90125158)

[3.3. Research Flow 19](#_Toc90125159)

[3.4. Personality Prediction 20](#_Toc90125160)

[3.4.1. Dataset Split 20](#_Toc90125161)

[3.4.2. Model 20](#_Toc90125162)

[3.5. Matching Algorithm 22](#_Toc90125163)

[3.6. Legal, Ethical, Social and Professional (LESP) Issues Analysis 23](#_Toc90125164)

[Chapter 4 24](#_Toc90125165)

[3. System Requirement Specification 24](#_Toc90125166)

[3.2. Chapter Overview 24](#_Toc90125167)

[3.3. Requirements 24](#_Toc90125168)

[3.3.1. Functional Requirements 24](#_Toc90125169)

[3.3.2. Non-Functional Requirements 25](#_Toc90125170)

[3.4. Use Case diagram for Dating application 25](#_Toc90125171)

[3.5. Sequence diagram for Dating application 26](#_Toc90125172)

[3.6. Component diagram for Dating application 27](#_Toc90125173)

[Chapter 5 28](#_Toc90125174)

[4. Implementation / Designing 28](#_Toc90125175)

[5.1. Mobile App 28](#_Toc90125176)

[5.1.1. Flutter Framework 28](#_Toc90125177)

[5.2. Chatbot 33](#_Toc90125178)

[5.2.1. Dialog Flow 33](#_Toc90125179)

[5.3. Back-End Server 34](#_Toc90125180)

[5.3.1. Firebase 34](#_Toc90125181)

[5.3.2. Model 36](#_Toc90125182)

[Chapter 6 40](#_Toc90125183)

[6. Result 40](#_Toc90125184)

[6.1. Introduction 40](#_Toc90125185)

[6.2. Testing 40](#_Toc90125186)

[6.2.1. User Acceptance Testing 40](#_Toc90125187)

[6.2.2. Test Cases 41](#_Toc90125188)

[6.2.2.1. Splash screen 41](#_Toc90125189)

[6.2.2.2 Login Screen 42](#_Toc90125190)

[6.2.2.3. Chatbot Screen 42](#_Toc90125191)

[6.2.2.4. Matching Chat Screen 42](#_Toc90125192)

[6.2.2.5. User profile Screen 43](#_Toc90125193)

[6.3. Experiment 44](#_Toc90125194)

[6.3.2. Binary Classification on Essays 44](#_Toc90125195)

[Chapter 7 45](#_Toc90125196)

[7. Concluding Remarks 45](#_Toc90125197)

[7.1. Accomplishment of the research objectives 45](#_Toc90125198)

[7.2. Problems encountered 46](#_Toc90125199)

[7.3. Research contributions review 46](#_Toc90125200)

[7.4. Future Work 46](#_Toc90125201)

[References 47](#_Toc90125202)

[Appendix A App Component Structure 50](#_Toc90125203)

[Appendix B Research Process Flow Chart 51](#_Toc90125204)

[Appendix C Mobile User Interface 53](#_Toc90125205)

[Appendix D Example Dating Questions 54](#_Toc90125206)

[Appendix E Admin Dashboard 55](#_Toc90125207)

[Appendix F User Survey Questionnaire 56](#_Toc90125208)

[Appendix G Responses for User Survey Questionnaire 59](#_Toc90125209)

[Appendix H User Evaluation Questionnaire 62](#_Toc90125210)

[Appendix I Gantt Chat 65](#_Toc90125211)

List of Figures

[Figure 1 : Rich picture 3](file:///E:\NSBM\New%20folder\renew-chapter\renewchapter.docx#_Toc89205707)

[Figure 2 : Feedforward neural network (source: deepai.org) 13](file:///E:\NSBM\New%20folder\renew-chapter\renewchapter.docx#_Toc89205708)

[Figure 3 : Sigmoid activation function (Wikipedia) 14](file:///E:\NSBM\New%20folder\renew-chapter\renewchapter.docx#_Toc89205709)

[Figure 4 : Recurrent neural network (source : Abhijeet Kamble 2019) 16](file:///E:\NSBM\New%20folder\renew-chapter\renewchapter.docx#_Toc89205710)

[Figure 5 : Bidirectional LSTM with an attention layer 21](file:///E:\NSBM\New%20folder\renew-chapter\renewchapter.docx#_Toc89205711)

[Figure 6 : Use case diagram 25](file:///E:\NSBM\New%20folder\renew-chapter\renewchapter.docx#_Toc89205712)

[Figure 7 : Sequence diagram 26](file:///E:\NSBM\New%20folder\renew-chapter\renewchapter.docx#_Toc89205713)

[Figure 8 : Component diagram 27](file:///E:\NSBM\New%20folder\renew-chapter\renewchapter.docx#_Toc89205714)

[Figure 9 : Main.dart 29](#_Toc89205715)

[Figure 10 : Chat screen 30](#_Toc89205716)

[Figure 11 : Loging screen 30](#_Toc89205717)

[Figure 12 : build.gradle 31](#_Toc89205718)

[Figure 13 : Profile screen 32](#_Toc89205719)

[Figure 14 : Build.gradle in app 32](#_Toc89205720)

[Figure 15 : Intent path for all requests by intent 34](#_Toc89205721)

[Figure 16 : Firestore database chats field 35](#_Toc89205722)

[Figure 17 : Firestore database users field. 35](#_Toc89205723)

[Figure 18 : Import libraries 36](#_Toc89205724)

[Figure 19 : Function of the loading dataset 37](#_Toc89205725)

[Figure 20 : Display dataset 37](#_Toc89205726)

[Figure 21 : Generate word embeddings 38](#_Toc89205727)

[Figure 22 : Calculation functions 38](#_Toc89205728)

[Figure 23 : Embeddings and labels 39](#_Toc89205729)

[Figure 24 : Training function 39](#_Toc89205730)

# Chapter 1

# Introduction

## Relevance of the research problem

The internet today, the demand for free dating services has created a large industry for online dating. In 2020, the dating industry was worth 3.08 billion GBP and is increasing at a substantial rate (Statista 2021), with 20% of committed relationships begin online (eharmony.com 2015).

There is a major liability put on online dating platforms because of the increasing number of people using the internet to find their partners. Popular dating sites are currently focusing on gamification rather than enhancing match quality (Rocha & Fakultet 2018). This causes biases in dating services, favoring a small percentage of users while negatively impacting minority users (Chaney et al. 2018). In addition, there is frequently an excess of choice in matches (Iyengar & Lepper 2000), with the majority of users' time spent looking for suitable companions. Personality plays an important role in predicting the quality of a relationship (Robins et al. 2000, Jarrett 2018).

Within the area of Linguistics and Computer Science, Natural Language Processing (NLP) is the study into using computers to read, analyse and derive meaning from human language. There have been vast improvements in the field of NLP through increased computational power and deep learning with neural networks. Automatic personality prediction from language is one of many text classification problems within NLP. Predicting personality from text is more difficult than other sources, such as videos or audio (Yang & Glaser 2017). That is not to say the limits of predicting personality through text have been reached.

Personality can be quantified using the Big Five personality traits which describes a user's personality along five dimensions: Agreeableness, Conscientiousness, Extroversion, Openness and Neuroticism.

## Motivation

* To using popular dating platforms do not always use it as a factor for matching. This research aims to show matches can be generated with personality similarity as the deciding factor.
* To achieve the goal of matching users based on personality, automatic personality prediction is used. This is the task of identifying a person's personality given a piece of text they provided.

## Aim

To build a user interface to interact with the chatbot and talk to matches with using an NLP based approach to create a matching algorithm for a dating application, where personality is used as a means of matching users

## Problem Statement

Popular dating systems do not always use it as a factor for matching. It is hard to find their partners because there is often too much choice in matches and also the majority of users’ time is spent finding potential matchings.

## Research Question

The research is and the objectives of the research are entirely built on following research problems.

* How to show matches can be generated with personality similarity as the deciding factor?
* How identifying a person’s personality given a piece of text they provided?

## Research Objective

The aim of this research is to show matches can be generated with personality similarity as the deciding factor. Major outcome of the service developing in this research is that where personality is used as a means of matching by that given a piece of text they provided

* To identify that the given a text and matching users based on personality, automatic personality prediction is used.
* To analyze their personality to design a chatbot that is capable of collecting data from an individual to later
* To implement model NLP based approach to create a matching algorithm and develop a user interface to interact with the chatbot and talk to matches.
* To evaluate to test the accuracy of the personality model on the data.

## Proposed Rich Picture

A Rich Picture is a method of exploring, acknowledging, and defining a situation and then expressing it through diagrams to form a rough mental model. A detailed graphic aid in starting a conversation and reaching a wide, common knowledge of the topic.

Diagram

Description automatically generated

Figure : Rich picture

## Challenges and Limitations

* Chatbots are not very common within the dating industry.
* Hard to find suitable data set (Found a essay type dataset , This dataset consists of 2468

essays written by psychology students. The students were asked to write anything for

20 minutes and complete a Big Five Inventory questionnaire. (Pennebaker & King

1999)

* There is no perfect dating algorithm exists due to the complexities and differences

between people and cultures. This is best known in mathematics as the Stable Marriage

Problem (Shiet al. 2018).

* A profile matching algorithm surveys users to learn about their preferences for a partner

such as height, age or interests. If two users have characteristics they both liked in each

other, they will be matched.Match.com and eHarmony announced they used preference

matching to pair users (Kessler 2011, Knapton 2017).

* Understanding of Word Embeddings

# Chapter 2

# Literature review

## Introduction

The purpose of this chapter is to demonstrate the many ways in which a chatbot may be built, as well as the various categories and descriptions of the measures used to assess bot performance. This helps in the creation of more effective bots. and reviewing the way it is used in dating algorithms in the previous research article. The dating platform's way of using the profile matching technique on Match.com is well reviewed in this study, and the collaborative filtering approach is mostly used today on online platforms. Here's more on the dating platform's collaborative filtering approach with the Elo Score concept. The concept of the "Big Five" personality traits comes from psychology and refers to five basic personal characteristics. Openness, conscientiousness, extraversion, agreeableness, and neuroticism are the Big Five personality traits. The basic structure of all personality traits is said to be represented by these five elements. The personality classification using LIWC, and the neural networks used in each model will be examined in detail.

## Chatbots

Chatbots are unusual within the dating platform. ‘Match.com’ a dating assistant called Lara chatbot as a companion for their dating service (Match.com 2018) .The Lara chatbot offers date recommendations, daily matches, and dating advice. Aside from Lara, no other dating company has employed a chatbot to evaluate individuals as part of their system. Although the chatbot is a tool for gathering user data for personality assessment, it is not the subject of this study. A brief overview of the types of chatbots (Jwala etal. 2019) is handed to help understand chatbots current state and its research introducing the development of chatbot has turn trendier and so far several conversational chatbots were organized which replaces the traditional chatbots. A chatbot is a computer program which is used to interact with humans and fulfill their needful. Chatbot gives the response for the druggie query and sometimes they're able to execute tasks also, Because of the wide availability of development platforms and resources, early chatbot creation was extremely tough, but recent chatbot development is considerably easier. Natural Language Processing (NLP) or Deep Learning can be used to create a chatbot. As a result, a large amount of data must be trained. Bots created with Deep Learning are more advanced than regular chatbots.

### Rule-Based Chatbots

The most popular chatbots employ a rule-based approach, in which a discussion proceeds according to a pre-determined path. They're best suited to chatbots with a specific objective in mind. Pattern matching and other techniques are utilized to create these chatbots. This makes them incredibly simple to construct, but the user experience suffers as a result. Artificial Intelligence Markup Language (AIML) is an XML-based language for writing chatbot rules (Das Gracas et al. 2013). Because AIML uses such a theoretical framework in its syntactic and semantic structures, this research study focuses on Pattern Recognition. After that, the AIML language is explained, followed by an application example for each AIML command/tag. Also demonstrated is the use of AIML embedded tags to handle sequence conversation constraints between people and machines. Finally, computer systems that aid in the creation of AIML-based chatterbots are classified and described.

Rule-based chatbots provide a much more regulated user experience since they are predictable and simple to create. When a user requests information or asks a question that does not follow the conversational flow, problems arise. Rule-based chatbots are useful for repeated tasks, but more complicated needs necessitate the use of more advanced methods.

### Generative Chatbots

Instead than following a set conversational flow, generative chatbots use machine learning models to have human-like interactions. Instead than coming from a fixed set of predetermined responses, answers are created. Because generative chatbot models parse a user inquiry word by word, they are more prone to mistakes in spelling and grammar. This necessitates a higher level of accuracy in their training. Microsoft Tay is a prominent example of such a chatbot (Liu 2017). This article discusses the Microsoft AI chatbot, which was introduced in March 2016. We believe Tay's example exemplifies an issue with the nature of learning software (LS) that interacts directly with the public, as well as the developer's role and duty. We argue that when LS interacts with people directly or indirectly through social media, the developer bears extra ethical obligations beyond those that apply to ordinary software. There is also the added responsibility of caring for others.

### Retrieval-Based Chatbot

A retrieval-based model is a cross between generative and hybrid of rule-based chatbots. A user's question and context are matched to an intent that the chatbot has predefined. The chatbot can then choose from a list of responses how to handle the intent. When a user asks a chatbot, "What's the weather today?" the chatbot will recognize the input as a weather intent and answer accordingly. This model is less complex and more predictable than a generative chatbot. In addition, they are more adaptable to user input than rule-based chatbots. As a result, the majority of consumer chatbots are retrieval-based.

## Dating Algorithms

Because of the complexities and variances between people and cultures, there is no such thing as a flawless dating algorithm. In mathematics, this is known as the Stable Marriage Problem (Shi et al. 2018). This paper looks at a more generalized version of the stable marriage problem, in which varying numbers of men and women must be matched pairwise and the appearance of single men and women is unavoidable. Theoretical analysis and numerical simulations show that even a small deviation from the equality condition in the number of men and women can have a significant impact on the Gale-Shapley algorithm's matching solution. These findings shed light on a variety of real-world issues. Compromises must be made when deciding on the factors included in a dating algorithm.

### Profile Matching

Users are polled to learn about their preferences for a relationship, such as height, age, and interests, using a profile matching algorithm. Two users will be paired if they share characteristics that they like in each other. Preference matching was used by Match.com and eHarmony to link users (Kessler 2011, Knapton 2017). When a user's choices and activity on the service did not match, both platforms detected problemsFor example, 57 percent of women who said a partner's willingness to have children is a "must-have" on the website messaged partners who didn't meet the criterion. As a result, their recommendation algorithms adjust for criteria that can be compromised, such as recommending partners with no desire to have children.

### Collaborative Filtering

Collaborative filtering is used by many systems to recommend things on e-commerce websites, movies on streaming platforms, and individuals to follow on social media. It is used in the dating world to recommend matches to other users. The process of filtering items using the opinions of others is known as collaborative filtering (Schafer 2007). It is founded on the notion that people who enjoy similar things like similar things. For collaborative filtering, a variety of machine learning approaches can be utilized, including clustering algorithms, matrix factorization (Mustafa et al. 2017), and, more recently, neural networks (He et al. 2017). This method has been successfully applied to dating algorithms (Krzywicki et al. 2015, Brozovsky & Petricek 2007). However, data suggests that collaborative filtering homogenizes user behavior and lowers the algorithm's utility (Chaney et al. 2018). Due to popularity bias, this type of algorithm can be troublesome. Users that have a lot in common with others are always recommended, resulting in a small number of well-known people. Furthermore, early users of the system have a significant influence on the recommendation system's future decisions. If an early user dismisses a profile with a particular feature, a comparable person will not be offered the profile because it is expected that they will dislike it as well.

### 2.2.3. Elo Score

Tinder is a popular mobile dating software that shows users a stack of profiles and asks them to swipe right to signal interest or swipe left to dismiss the profile. If two users swipe right for each other, they will be matched and able to communicate within the app. Tinder's most essential feature is its algorithm for sorting profiles. Tinder acknowledged that they previously employed an Elo ranking system, despite the fact that many dating sites do not share their algorithms (Tinder 2019). This is widely used to assess chess players' abilities (Tiffany 2019), but it has also been applied to sports, video games, and board games. Elo assigns a score to Tinder users, with the greater the score, the more attractive they are. When other users swipe right on a person's profile, their score rises; if they swipe left, their score falls. If the user swiping has more points, the effect on an individual's score is greater. The mathematical model of the Elo rating system demonstrates this (Aldous 2017). Each user is given a starting score, which is a real number yi. When user I comes across user j's profile, a function Y is used to update both players' ratings :

if i beats j then yi → yi + Υ(yi − yj ) and yj − Υ(yi − yj )

if i loses to j then yi → yi − Υ(yj − yi) and yj + Υ(yj − yi)

## Predicting Personality

The success of a relationship is heavily influenced by one's personality (Donnellan et al. 2004, Robins et al.2000). Having a personality that is similar to your partner can indicate satisfaction in a relationship (Laubu et al. 2016, Wang et al. 2018).

### Big Five Personality Model

The Big Five personality traits are the most recognized and well-researched global model for personality in psychology. For defining a human's personality, the model employs five distinct dimensions.

These are openness, conscientiousness, extroversion, agreeableness and neuroticism.

The Big Five traits can be described as follows (Donnellan et al. n.d.):

* Openness

Openness refers to the width and depth of one's existence, as well as the uniqueness and complexity of one's experiences. Individuals with a high level of openness are more informed, insightful, and analytical, and they seek out new experiences. They are also more artistic and investigative.

Facets: curious, excitable, unconventional , wide interests, artistic , imaginative

* Conscientiousness

Conscientiousness that the ability to control impulses to facilitate goal-directed behaviour. Those high in this trait follow norms and rules, and are efficient in planning, organizing and prioritizing tasks.

Facets: Efficient, thorough, organized, not careless, not impulsive, not lazy

* Extroversion

Extroversion refers to a person's ability to be lively and enthusiastic, and it involves characteristics such as friendliness, assertiveness, confidence, and ambition.

Facets: outgoing, energetic, forceful, sociable, enthusiastic, adventurous

* Agreeableness

The person's level of altruism, cooperation, willingness to conform to group standards, and warmth or friendliness are all agreeable.

Facets: forgiving, sympathetic, not stubborn, warm, not showoff, not demanding

* Neuroticism

Neuroticism is a personality trait that compares emotional stability with anxiety, uneasiness, and depression. Self-conscious, temperamental, impulsive, and stressed people have high levels of this feature.

Facets: shy, not self-confident, moody, not contented, irritable, tense

The Big Five personality classification model is extensively used in personality classification and will be used to classify people for the dating application.

### Essays Dataset

The Essays dataset is used in the majority of personality prediction studies (Pennebaker & King 1999). There are 2468 essays authored by psychology students in this collection. The students were given 20 minutes to write whatever they wanted and to complete a Big Five Inventory questionnaire. Students were asked to rate themselves on a five-point scale based on a list of descriptions in the survey. Essays are associated with each of the Big Five personality traits in the final dataset. Each trait is either labeled ”y” to indicate that the student has the trait or ”n” to indicate that the student does not have the trait. The five personality qualities were shown to be correlated with language dimensions, implying that personality traits might be inferred from texts.

### Linguistic Inquiry and Word Count

Linguistic Inquiry and Word Count (LIWC) is a popular text analysis tool. The program counts the number of words that match linguistic markers that are present (Conglomerates 2015). The word cried, for example, falls under the headings of sadness, negative emotion, overall affect, verb, and past focus. Percentages can be calculated based on the amount of words in each category to determine how dominant a piece of text is within that category.

Correlations between these linguistic markers and personality traits can be discovered using LIWC. Using approaches such as support vector machines, naive Bayes, decision trees, and linear regression, LIWC was used to develop models for the Essays dataset (Mairesse et al. 2007, Yarkoni 2010, Tighe et al. 2016).

### Word Embeddings

Word embeddings are a common method of encoding words in a document as a set of multidimensional vectors (Mikolov et al. 2013). The semantic meaning of a word can be captured by a vector for that word. Words with comparable semantic meanings are grouped together in the same space. The vectors for the words "car" and "bus," for example, may be comparable, while "car" and "elephant" would have completely different vectors. Using word embeddings to capture the semantic content of a sentence extracts linguistic markers, similar to LIWC. The vectors can be utilized in a machine learning model for additional analysis. BERT (Bidirectional Encoder Representations from Transformer) is the most recent breakthrough for creating word embeddings (Devlin et al. 2019). By increasing the scale of the neural network, RoBERTa, a version of BERT, was built with superior performance (Liu et al. 2019). The workings of BERT and RoBERTa are detailed in further detail in the background section.

### Neural Networks

Machine learning models inspired by biological neurons in the brain are known as neural networks. They're useful since they can learn non-linear patterns in data. They accept input and, after a series of calculations, return an output via their hidden layers. For personality classification, for example, the input may be a list of word embeddings, and the output could be a likelihood of a personality attribute being present.

Diagram

Description automatically generated

Figure : Feedforward neural network (source: deepai.org)

During the training phase of a neural network, the weights and biases of the neural network are learned in order to produce the desired output. The weights and biases of a neural network are normally created at random when it is first set up.

A neural network is made up of layers of nodes. Each node accepts a set of numerical inputs and a set of weights of equal size. A bias is created by adding the dot product of the input and weights. Finally, the number is applied to a function that activates something. The neural network can learn non-linear functions from the input data using the activation function. A binary step activation function is a simple example, where a value greater than a threshold should output one and else zero. Complex activation functions, such as the sigmoid , are beneficial for binary classification because they convert any integer to a real value between 0 and 1.

A picture containing diagram

Description automatically generated

Figure : Sigmoid activation function (Wikipedia)

During the training phase of a neural network, the weights and biases of the neural network are learned in order to produce the desired output. The weights and biases of a neural network are normally created at random when it is first set up. A dataset comprising data and labels is required to train a neural network. A list of texts and labels for each piece of text, such as personality trait classifications, will be included in the data for a text classification task. Text is frequently converted to word embeddings in the field of natural language processing (NLP). Each record in the dataset's created word embeddings is fed into the neural network one by one, and the output is compared to the dataset's true value. The difference between the neural network's output and the true value from the dataset is measured using a loss function. The loss function produces a number; the smaller the number, the better the model performed.

Because of the loss, the model's weights will need to be adjusted in order to improve its performance. The chain rule is used to compute the derivative of the loss function with regard to a node's weights in a method known as backpropagation. The weights' gradients are the derivatives. An optimiser's purpose while employing gradients is to adjust the weights using a process called gradient descent, in which the weights change to reduce the gradient to the global minimum. This is the point at which the weight reduction is at its smallest potential level. The optimizer is given a learning rate, which is used to indicate how often weights should be updated.

Training will take too long if the learning rate is too low, but if the learning rate is too high, the model may overshoot the global minimum. For optimal training, the optimizer's learning rate must be modified.

When done one by one for each record in the dataset, calculating the loss, gradients, and then optimizing the model takes a long time. A batch of records from the dataset is usually trained at once to improve training speed. After the batch, the optimizer totals the loss for each record and updates the model.

Typically, neural networks are trained across a number of epochs, with each epoch representing one iteration of the dataset being fed through the model. The dataset is usually divided into two parts: training and testing. The model learns using the training set but never sees the test set during training. The test set is used to evaluate the model using data that has never been seen before. The model's weights are not updated throughout testing, therefore there is no training. Overfitting occurs when a model performs well on the training set but not so well on the test data. Splitting the data into two sets allows you to see if the model is generic enough to handle any input or if it is only good for the training set.

### Convolutional Neural Networks

Convolutional neural networks (CNNs) are widely used in computer vision (Krizhevsky et al. 2012), but they also have applications in natural language processing. Convolutional neural networks (CNNs) are neural networks with at least one convolutional layer. This layer connects nodes in the previous layer that are close to one other to one node in the convolutional layer. This is analogous to words close each other being fed into a single common node in NLP. This method offers the advantage of extracting only the most important attributes from a set of words while discarding irrelevant data. Additionally, pooling layers are prevalent in CNNs, which take several nodes as input and return the maximum or average value. The pooling layers, like the convolutional layer, lower the model's complexity and prevent overfitting by generalizing features. For personality prediction, CNNs have been thoroughly explored. On the Essays dataset, personality traits were categorized (Jiang 2018, Majumder et al. 2017, Mohammad & Kiritchenko 2014), exceeding earlier LIWC results.

### Long Short-Term Memory Neural Networks

A recurrent neural network (RNN) is a loop-containing neural network. This is when a node's output is used as an input for a node farther down the network. RNNs are useful for sequenced data, like as sentences, because each item in the sequence has a better grasp of the items before it when the items before it are provided.

Diagram

Description automatically generated

Figure : Recurrent neural network (source : Abhijeet Kamble 2019)

In theory, RNNs can leverage previous context in a sequence to aid understanding of the present item. In fact, the RNN is unable to learn when essential features are further back in the sequence (long-term dependencies) (Bengio et al. 1994). LSTMs (Long Short-Term Memory Neural Networks) are neural networks that are specifically intended to learn long-term dependencies more efficiently (Hochreiter & Schmidhuber 1997). A cell state is transferred between nodes that can store context within an LSTM layer. The beginning cell state and output from the first node in the chain are transferred down to the last node in the chain, forming an LSTM. This is the same as reading a statement from beginning to end while keeping the context of the previous words. Three gates are required for each LSTM node. The first is a forget gate, which determines whether or not the cell state should be removed if the context is irrelevant. The second, an input gate, determines if the cell state should be changed and, if so, updates it. The output gate is the last gate, and it generates an output using the input and the updated cell state. The output as well as the state of the cell are transferred to the next node. This process is repeated until the output of all nodes in the layer has been determined. The network's last node produces a context vector that can be utilized for categorization later on.

The context vector of a conventional LSTM is created by reading a sentence from beginning to end. To improve semantic understanding, a bidirectional LSTM (BLSTM) creates a second context vector by reading the sentence backwards. To do this, an additional sub-layer of the LSTM is created, with the LSTM beginning at the end of the input. The forward and reverse context vectors are concatenated in the final context vector produced by the BLSTM. This method is useful when the context of a word is determined after it has been placed in a phrase.

For example, the word "bank" in "money bank" gets its meaning from the preceding phrase, whereas "bank of the river" gets its meaning after it has been placed in the sentence. Long-term dependencies are still a problem for LSTMs, but to a lesser extent than RNNs (Bahdanau et al. 2015). This is because putting the entire output of the LSTM layer into a context vector results in the loss of critical information. In lengthier sentences, this is extremely challenging. The employment of an attention mechanism, which was first proposed for text translation, is one possibility (Bahdanau et al. 2015). Rather than transmitting its output to a neighboring node, each node in an LSTM layer additionally passes it to an attention layer. As a result, rather than simply the context vector, the attention layer receives the output of all the nodes in the LSTM layer. The attention layer determines what information is significant by utilizing a softmax activation function to calculate a probability for each LSTM node's output. To reduce the values of insignificant inputs, the probability is multiplied by the output of the LSTM layer. In compared to CNNs, LSTM networks have not proven as successful at predicting personality. On the essays dataset (Jiang 2018), a BLSTM performed marginally better with certain qualities, but adding an attention mechanism to the BLSTM improved outcomes on all traits, beating CNN models with and without attention.

### BERT and RoBERTa

Google's BERT (Bidirectional Encoder Representations from Transformers) neural network is inspired by the Transformer design (Devlin et al. 2019). BERT has grown in popularity as a result of its ability to perform state-of-the-art results on a variety of NLP tasks, particularly text categorization tasks. BERT consists of 12 encoders, each with its own self-attention layer. Unlike an LSTM, which receives input from beginning to end, BERT reads the full input sequence at once and uses the self-attention layer to learn the context of a word from its surrounding words. BERT was trained using two tasks. Masked Language Modeling (MLM) and Next Sentence Prediction were the methods used. During MLM, the token [MASK] or a random word was substituted for 15% of the words in a sequence. This model was required to anticipate the words that had been replaced. BERT was trained with Next Sentence Prediction by giving it two sentences and asking if the second sentence will be the next sentence after the first. RoBERTa is a neural network that is similar to BERT but has additional parameters, allowing it to build larger word embeddings with more contextual information for each word (Liu et al. 2019). RoBERTa outperforms BERT on most tasks, although it necessitates more computer power and memory due to its larger size. The language models BERT and RoBERTa can be used for two purposes. The initial step is to create word embeddings to feed into a different model. The second step is to fine-tune the model such that it can learn a text categorization task. This entails adding an untrained layer to the model's end and training the entire thing for a few epochs.

On the Essays dataset, the most recent advancement in personality categorization takes a pre-trained RoBERTa model and fine-tunes the output by adding an untrained layer of neurons to the network's end (Jiang et al. 2019). RoBERTa was able to improve on all personality variables except conscientiousness by training the model using the Essays dataset.

# Chapter 3

# Methodology

## 3.1. Introduction

The following sections describe the data collection procedure and methods used to develop the system. The dataset how to split in personality prediction and using in model BLSTM with an attention mechanism and the word embeddings. And also needs that are both functional and non-functional was designed in order to gain a better understanding of what the system should provide to the user. This section include about matching people using matching algorithm, the chatbot and backend-server.

## Methodological approach

There were 7.9 million social media users in Sri Lanka in January 2021 (Digital 2021) and now a day teenagers very usually use the social media platforms, Submitting survey questionnaire through the platform to find out more about nowadays up running dating services and thoughts about dating platforms. That’s very helpful for to do this research and what they desire from the those platforms. and the dataset is used in the majority of personality researches .Also focus researching other authors' published works and experiment of using other dating applications well and discuss among a group of colleague about a topic to gather features that can be used for further research .So the best appropriate approach is mixed method approach.

## Research Flow

The flowchart should give you a solid concept of how to perform library research in the right order. Refer to the tabs above for more information on any step or ask a Librarian. Using flow charts to organize data and then communicate conclusions using flow charts to take the reader through the findings [Appendix B]. A flow chart is a diagram that depicts the operations and motions of a complex system. The research process flow charts serve as a guiding map to help us get from the beginning of the research to the end of a conclusive understanding.

## Personality Prediction

### Dataset Split

With 10-fold stratified cross-validation, the dataset will be partitioned (Kohavi 1995). Cross-validation divides the dataset into ten folds at random and trains the model on nine of them. Testing is done on the last fold. This is normally repeated ten times on all conceivable fold combinations, with the average results used. Because training neural networks takes a long time, only one fold combination is used, greatly speeding up training.

Due to its random nature, cross-validation does not guarantee that each fold evenly represents all of the classes in the dataset. This indicates having an equitable split of 'y' and 'n's for a personality attribute when it comes to personality prediction. To avoid this issue, stratified cross-validation makes sure that each fold contains an equal number of classifications.

When compared to a traditional train-test split, cross-validation allows the model to learn the entire dataset. This is especially helpful when dealing with smaller datasets. It also migrates the effect of overfitting, which is a major problem with neural networks (Zaremba et al. 2015). The dataset is randomized using a constant seed value, resulting in the identical splits being created while training and testing the model.

### Model

For each of the Big Five personality traits, five models were created. The model chosen for the task was a BLSTM with an attention mechanism. It is a good alternative because it may draw context from both directions in a sentence (Figure 5). The model's attention mechanism ensures that it can learn long-term dependencies and select the most significant phrases for personality trait classification (Jiang 2018). To construct word embeddings, previous models used word2vec (Majumder et al. 2017) and fastText (Jiang 2018). There hasn't been any research done on the use of context-sensitive word embeddings to convey more information. As a result, the model will be used to test both BERT and RoBERTa word embeddings.

Chart, box and whisker chart

Description automatically generated

Figure : Bidirectional LSTM with an attention layer

The neural network's first layer uses either BERT or RoBERTa to convert each word in a sentence to word embeddings. The LSTM layer captures the sentence's meaning from beginning to conclusion and vice versa. To prevent overfitting the training data, a second LSTM increases model complexity. The attention layer receives the embeddings and outputs from both LSTM layers. The most significant information is chosen for classification by the attention layer. The last layer consists of only one node, which receives the output from the attention layer and uses a sigmoid activation function to return a probability from 0 to 1. The trait exists if the probability is greater than or equal to 0.5. Otherwise, the trait isn't present.

The binary cross-entropy loss function was chosen because predicting personality traits is a binary classification problem. The difference between two probability distributions, in this case, the predicted values from the model and the true values from the dataset, is quantified by this loss function.

Adaptive Moment Estimation (Adam) was the optimizer employed (Kingma & Ba 2014). For each parameter in the network, Adam utilizes distinct adaptive learning rates. Individual learning rates function best in situations where gradients are sparse. When a neural network doesn't get enough data to tweak its weights, this happens. Sparse gradients are typical in NLP, which is why Adam was chosen as the model's optimizer.

Overfitting is a risk with LSTMs (Zaremba et al. 2015). The model performs well on the training set but not so well on the validation set. The model should acquire generalized features that appear in both the training and validation sets, not only the training set, to avoid overfitting. Dropout and weight loss are two methods that can be utilized to achieve this. Dropout is the process of disregarding nodes in a neural network at random in order to avoid co-adaptation. This is where one node is taught to repair another node's faults. Co-adaptations do not generalize to new data, resulting in overfitting. However, dropout accounts for this issue. Dropout is added to the first LSTM layer in the model. When the dropout is 0.2, it indicates that 20% of the nodes in the first LSTM layer are disregarded. Overfitting is usually indicated by significant weight values. Nodes with large weights are penalized by weight decay. When updating the weight of a node, it does so by deleting a part of the weight. As a result, their weights do not reach extremes of positivity or negativity.

Early stopping, a technique for stopping training when no learning occurs, was used to speed up training. Training is terminated if the validation loss does not reduce after a certain number of epochs. There isn't likely to be a better answer with more training.

## Matching Algorithm

The deciding criterion for pairing users was personality similarity. This is because similarities in a relationship might indicate happiness (Laubu et al. 2016, Wang et al. 2018). It's calculated by comparing the number of Big Five personality traits shared by two users on a scale of 0 to 1. The greater the value, the more users a couple has in common.

## Legal, Ethical, Social and Professional (LESP) Issues Analysis

Table : LESP analysis

|  |  |
| --- | --- |
| Legal | * Violation of The Computer Misuse Act of 1990 * Violation of the General Data Protection Regulation (GDPR) * Violation of the Data Protection Act of 2018.   The data used in this study and prototype is publicly available, non-sensitive, and has no legal implications. |
| Ethical | * The generation of offensive recommendations will be avoided by using a domain that is well-known and broadly approved. * Implementing a grammatically accurate rule base for knowledge base verbalization with data collected from reputable sources would prevent the generation of illegible or incorrect recommendations. * On the application, users will not be subjected to discriminatory validation based on disabilities, race, religion, culture, or language. |
| Social | * Recommendation It is preferable to deploy the system on a solid server with minimal downtime from a reputable cloud service provider to avoid the system being unreachable on a regular basis. * User issues should be addressed as soon as possible, and solutions should be provided. |
| Professional | * Users who access the application will not be subjected to any validations. Furthermore, the system can be categorized as a globally accessible system because it is platform neutral and computationally cheap to run. |

# Chapter 4

# System Requirement Specification

## Chapter Overview

In this chapter describes mainly specification of system requirements and structure of the component diagram. A system's requirements are embodied in a structured collection of information that requirements assessment is a critical procedure that ensures a system's success.

## Requirements

A list of functional and non-functional needs was created to better understand what the system should deliver for the user.

### Functional Requirements

* The user should be able to establish an account using their Google account when the app first runs.
* They must be able to sign in and out of their account once it has been setup.
* If a user creates a new account, they should go through an onboarding process to learn how to use the app. In addition, the user should be able to add information to their dating profile, such as their initial name, gender, and a profile image.
* A list of matches should be provided, including their profile image and name
* The user should be able to send messages to their matches
* The user should be able to communicate with a chatbot and see their chat history
* The user's profile, including their name, profile photo, and inferred personality attributes, should be visible to them.
* The user should be allowed to log out of the program.
* When a match is identified, the user should receive a notification.

### Non-Functional Requirements

* The user should not feel any stuttering or sluggishness when using the program.
* If the user tries to do an action that requires connectivity while offline, the app should warn them that they are not online.
* All network requests should be completed within a reasonable timeframe.
* The user should be able to open the app and immediately begin conversing with the chatbot.
* Unless there are uncontrollable causes, such as an inevitable operating system malfunction, the program should never crash while in use.

## Use Case diagram for Dating application

A use case diagram can summarize the details of the system's users (also known as actors) and their interactions with the system in the Unified Modeling Language (UML). An effective use case diagram depicts the scenarios in which your system or application interacts with people, organizations, or external systems, as well as the goals that your system or application assists those entities (known as actors) in achieving, as well as the scope of your system.

Diagram

Description automatically generated

Figure : Use case diagram

## Sequence diagram for Dating application

Sequence diagrams show interactions between classes as a series of messages exchanged over time. Event diagrams are another name for them. A sequence diagram is a useful tool for visualizing and validating the system's many runtime scenarios. These can aid in predicting how a system will behave and identifying responsibilities that a class may require during the modeling phase.

Diagram

Description automatically generated

Figure : Sequence diagram

## Component diagram for Dating application

Component diagram is a special kind of a UML diagram. Component diagrams are used to visualize a system's static implementation view. It depicts the arrangement of the components within a system, or the actual components of a system. Before the implementation, the components, such as libraries, files, and executables, must be arranged. At runtime, it represents the components of a system and It's helpful for system testing are some of the reasons for the component diagram's required.

Diagram

Description automatically generated

Figure : Component diagram

# Chapter 5

# Implementation / Designing

## Mobile App

A mobile app was chosen as the best alternative for delivering the service when compared to other options. If a user is given questions throughout the day [Appendix D], they are more likely to respond via an app rather than a website or other medium. It's also crucial to be able to message matches at any time. The system has to support both iOS and Android smartphones to match the service given by other dating applications. Building two distinct apps for both platforms is a huge undertaking that isn't doable in this project's time period.

Despite being a viable solution, the final product may be slow and lack the native-feeling user experience that one would expect from a mobile app. Flutter framework for creating cross-platform native apps, was the best answer. Flutter allows a developer to construct a native app by interacting with the native APIs of iOS and Android using JavaScript. Flutter allows for quick development while still delivering the quality and performance that users expect from a native app.

The user needs to login. If he or she does not have an account, they can create one very easily. After they login, the user can see the chatbot interface. (Appendix A and Appendix C) The user can chat with Lily and have fun with her. Following the prediction, Lily sends some profiles to the user, who can then chat with them via the match chat interface, which is located on the second tab. In the last tab, users can change their profile picture and can also add a bio to their profile. If the user clicks the log out button, they are taken to the login page.

### Flutter Framework

Flutter is a cross-platform UI toolkit created by Google. Flutter is compatible with Android, iOS, Windows, Linux, and Mac OS X. It's a free and open-source app development tools. Flutter develops apps using Dart, an object-oriented programming language. Dart's most notable features include a comprehensive standard library, async-awaiting, strong typing, garbage collection, and garbage collection. Dart shares many of Java's features and also makes use of those of other languages. Flutter developers can create native-like apps for any platform. It gives such a rich user experience because it employs the Dart programming language, which is quick, simple, and straightforward to compile into native code.

It greatly increases the application's overall performance when compared to alternative development platforms. These are the main reasons for chose flutter for mobile UI implementations.

A screenshot of a computer

Description automatically generated with medium confidence

Figure : Main.dart

The main() function is the Dart's top-level function. It is the Dart programming language's most important and vital feature. The main() function initiates the execution of the program. In a program, the main() function can only be used once [Figure 9].

A screenshot of a computer

Description automatically generated with medium confidence

Figure : Chat screen

Text

Description automatically generated

Figure : Loging screen

From simple "text" to "buttons" to "screen layouts," everything in a Flutter app is a widget in Flutter. The hierarchies are arranged in this widget so that they can be presented on the screen. Following figures shows some widget used in the application [ Figure 10,11,13]

Text

Description automatically generated

Figure : build.gradle

Gradle is an open source build system for automating the development, testing, and deployment of software. Scripts such as "build.gradle" can be used to automate processes. The Gradle build script, for example, can do a simple action such as copying files from one location to another before the actual build process begins [figure 12].

Text

Description automatically generated

Figure : Profile screen

Text

Description automatically generated

Figure : Build.gradle in app

## Chatbot

The chatbot Lily, whose objective is to learn about the user and propose matches, receives the majority of the user's attention in the app. Lily is mostly a rule-based chatbot, as the main focus of this research is on personality prediction. When a user first logs in to the app, Lily greets them and walks them through a pre-determined process to gather information for their profile. The user's name, age, gender, and profile picture are all included. Lily then goes over how the app works and what the user can anticipate from it.

### Dialog Flow

Dialog Flow is a service that provides everything you need to build voice and text-based conversational interfaces. It makes use of Machine Learning to have a dialogue that is as natural as possible. The entire process may be completed utilizing Dialog Flow’s web console's easy UI. There are several features that make Dialog Flow a compelling solution, including easy interfaces with multiple services, easy deployment on every platform (web, mobile), the possibility of adding a backend with some logic, and an integrated Small Talk functionality.

Diagram

Description automatically generated

Figure : Intent path for all requests by intent

## Back-End Server

By storing user data on a mobile development platform called Firebase, the time spent creating the back-end server was reduced. Firebase uses a NoSQL database to store user data. An API is used by the mobile app to communicate with Firebase.

### Firebase

Google Firebase is a mobile application development platform from Google that includes a number of useful capabilities for creating, managing, and improving apps. Firebase is essentially a set of tools that developers may use to create and expand applications based on demand [figure 17-18].

Firebase is designed to address three major issues that developers face: quickly creating an app with confidence, releasing and monitoring an app. Developers who utilize this platform gain access to services that they would otherwise have to design themselves, allowing them to focus on delivering rich application experiences.

Databases, authentication, push messages, analytics, file storage, and more are just a few of the notable capabilities of the Google Firebase platform. Developers can easily do on-demand scaling because the services are hosted in the cloud. Firebase is now one of the most popular app development platforms in the world.

Graphical user interface, text, application, email

Description automatically generated

Figure : Firestore database chats field

Graphical user interface, text, application, email

Description automatically generated

Figure : Firestore database users field.

A collection includes only documents. It can't have any raw fields with values in it, and it can't have any other collections in it. The names of the documents in a collection are all different. Provide private keys, such as user IDs, or Cloud Fire store can generate random IDs. There's no need to "add" or "remove" collections. The collection exists after creating the first document in it. A collection is no longer active if all of the papers in it are deleted [figure 16,17].

### Model

Using google Colab environment, free access to the "NVIDIA Tesla K80" GPU. The CUDA programming model is used to make optimum use of the GPU's many cores. The CUDA operations are significantly easier to run in Pytorch. The GPU greatly aids us in improving the performance of a sophisticated neural network model while also reducing the training time. The GPU has a huge number of smaller cores built in to assist with this operation. Setting pin memory = True leads in faster data transfer between the CPU and GPU when pushing a dataset loaded on the CPU to the GPU.

Graphical user interface

Description automatically generated

Figure : Import libraries

Many built-in modules in the Python Standard Library contain useful methods and data types for accomplishing specific tasks. Modules not included in the standard library can be used as well. A .py file contains a module [Figure 18]. Classes, functions, and other objects can be included in this file. A package is a collection of similar modules bundled under a single name. Use NumPy, SciPy, and Pandas etc.

Graphical user interface, text

Description automatically generated

Figure : Function of the loading dataset

A picture containing scatter chart

Description automatically generated

Figure : Display dataset

Graphical user interface, text, application

Description automatically generated

Figure : Generate word embeddings

Graphical user interface, text

Description automatically generated

Figure : Calculation functions

In below figure, [Figure 23] word embeddings allow you to create a dense representation of a word in which similar words have the same meaning (encoding). A dense vector of floating-point values is called an embedding.

Graphical user interface, application

Description automatically generated

Figure : Embeddings and labels

Scatter chart

Description automatically generated

Figure : Training function

.

# Chapter 6

# Result

## Introduction

The result of this study was obtained from matching people based on personality and interacting with them via chatbot and talking to matches and there also experiment of binary classification of an essay. The system's success was measur ed using one way. It was acceptance testing for users. Users were given the app to see if the initial requirements were completed and if any unforeseen benefits or problems were discovered.

## Testing

### User Acceptance Testing

User acceptance testing was a crucial indicator for evaluation because the purpose of this study was to create a dating service. Participants were shown a copy of the app as well as a questionnaire to complete. [Appendix H] contains a complete copy of the questionnaire. The user is guided through the app by the questionnaire. Creating a profile, signing in, answering the chatbot's common dating questions, finding a match, and messaging their matches are all part of the process. Participants were given actions to do in order to go in the app, and they were asked if they were successful. These actions were created to put the system's needs to the test.

User acceptance testing included six users, two of whom engaged with the app in person. Due to the events of the COVID-19 outbreak, the other participants utilized the app remotely and submitted their comments via skype call.

Many of the app's functions performed as intended, and participants noted no serious issues with them. Although there were no crashes, one participant reported delays in receiving messages from the chatbot. The difficulty was caused by a loss of internet access, however there was no reaction from the app regarding the issue.

The app's remaining comments focused on the user experience rather than the software. Several people had a great time chatting with the chatbot. The user interface's simplicity was also satisfied, proving that it was both easy and enjoyable to use.

Aside from the positive feedback, there was some criticism. One participant stated that conversing with a chatbot was difficult for them and that they would only respond with a few words. Others complained that the chatbot didn't do a good job of explaining the app's features. Finally, one participant noted that Lily had far too much responsibility. So over the matches that had been offered, they would have preferred to personally select matches and filter some of the users using their personality.

### Test Cases

Test cases are step-by-step procedures for verifying an application's or product's functionality. To write test cases to ensure that the application's functionality is working as expected according to the client's requirements. This includes both positive and negative test cases to ensure that the application's behavior is correct.

### Splash screen

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test case id | Test scenario | Test steps | Test data | Expected result |
| Tc-01 | Verify that the Splash screen does not crash when start the app | Check restarting app | Is screen load perfectly | Pass |
| Tc-02 | Verify that the Buttons and text are align and display properly | Check label alignments | Is label align properly | Pass |

### Login Screen

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test case id | Test scenario | Test steps | Test data | Expected result |
| Tc-03 | Verify that the login screen alignments properly | Check label alignments | Is label align properly | Pass |
| Tc-04 | Verify Email format | Check Email format | Is email format is correct | Pass |
| Tc-05 | Verify that user can login successfully | Check entering user’s email and password | Is email and password correct | Pass |

### Chatbot Screen

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test case id | Test scenario | Test steps | Test data | Expected result |
| Tc-06 | Verify that the chatbot screen alignments properly | Check label alignments | Is label align properly | Pass |
| Tc -07 | Verify that user can chat with chatbot | Check chatting with chatbot | Is chatting perfectly | Pass |

### Matching Chat Screen

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test case id | Test scenario | Test steps | Test data | Expected result |
| Tc-08 | Verify that the chatbot screen alignments properly | Check label alignments | Is label align properly | Pass |
| Tc-09 | Verify that user can see matches | Check cards view | Is can see matches properly | Pass |
| Tc-10 | Verify that user can chat with matches | Check chatting with matches | Is chatting perfectly | Pass |

### User profile Screen

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test case id | Test scenario | Test steps | Test data | Expected result |
| Tc-11 | Verify that the user profile screen alignments properly | Check label alignments | Is label align properly | Pass |
| Tc-12 | Verify that user can edit profile picture | Check can be edit profile picture | Is edit successfully | Pass |
| Tc-13 | Verify that user can edit bio | Check can be edit bio | Is edit successfully | Pass |
| Tc-14 | Verify that user can logout clicking by logout button | Check on click the logout button | Is logout successful | Pass |

## Experiment

### Binary Classification on Essays

The Essays dataset was acquired and parsed from a CSV (comma-separated values) file, which contained essays tagged with the authors' Big Five personality traits. A single essay that just had the line "Err:508" was eliminated from the dataset, leaving a total of 2467 essays. The essays were not further filtered because they all contained at least one emotionally charged word (Majumder et al. 2017). During training, if there were any neutral sentences in an essay, the attention mechanism should filter them out.

In an essay, personality traits were labeled with a "y" if they were present and with a "n" if they weren't. If a trait was present, these values were replaced with 1.0 and 0.0, respectively. The loss was calculated by comparing the model's output to the dataset after converting the tags to numbers.

To avoid the model misconstruing the identical words capitalized differently as two different words, the essays were transformed to lowercase text. For example, a lowercase essay with largely capitalized terms like "SHE WAS A REALLY COOL GIRL" might be evaluated differently. Maintaining capitalization consistency should help improve accuracy.

Stop words are frequently removed as part of the pre-processing of datasets for neural networks. "And," "is," and "the" are examples of common words. This has a negative impact on model performance. LSTMs, for example, require stop words to understand the semantic meaning of a phrase (Saif et al 2014). As a result, stop words were left in the essays.

Due to the complexity and scale of their models, generating word embeddings with BERT and RoBERTa takes a long time. When training is repeated across a number of epochs and embeddings must be regenerated, this becomes even worse. Word embeddings are generated once and used for each epoch to improve training speed. Before training, word embeddings were reduced to 400 tokens so that they could fit in memory. Bert word embeddings were tried first, followed by RoBERTa word embeddings.

# Chapter 7

# Concluding Remarks

## Accomplishment of the research objectives

The conclusions of the user acceptance testing reveal that using a chatbot has both benefits and drawbacks. Finding matches with a chatbot was a fun and personal experience that other dating sites don't offer. The app's simplicity was also praised, owing to the chatbot's role in the majority of the user's journey through the app. The personality prediction model would be hampered by one participant's difficulty communicating with the chatbot. The user's inferred personality would be inaccurate without enough quality data, and the matching algorithm would be less successful.

Despite the enjoyability of the user interface, the issue of the internet connection dropping while the participants were conversing with the chatbot was overlooked. If the user expects a response from the chatbot but there is no internet access, the chatbot waits till connectivity is restored before responding without informing the user. A better approach would be for the chatbot to state that they are unable to connect to the internet and that they should try again later. Another issue that can be resolved by reworking the onboarding experience to make it clearer is the chatbot's poor explanation of the app and one user requested to develop the admin dashboard. An admin web dashboard to search for users, matches, and test all of the scheduling tools. Appendix E contains a simple UI of a dashboard. React, a web framework, was used to create this its indicate only dummy data site for the preview.

The findings demonstrate that an attention based BLSTM is a good fit for the job of personality prediction. On extroversion, openness, and neuroticism, it outperformed the state-of-the-art fine-tuned RoBERTa model, as well as the state-of-the-art attention-based CNN on conscientiousness. The only trait on which the model performed poorly was agreeability.

## Problems encountered

The model's performance on the Essays dataset, the model's results from the questionnaire revealed that it can detect personality traits from unknown data. With a difference of less than 1%, the average accuracy of the Essays dataset and questionnaire answers were almost identical.

In the model training, there were parameters unknown to the model because of the versions used for technologies like scorch Pytorch technologies. This was difficult to achieve within the time and scope of the project.

## Research contributions review

Personality has been proven to be a deciding factor in dating apps, but the full degree of the system's performance will not be known until it is put to use in the real world. The attention-based BLSTM with RoBERTa word embeddings has proven to be an effective predictor of personality, but the training is incomplete. The usage of a chatbot to gather data from users has been shown to be effective. The chatbot allows users to have a more engaging experience when looking for matches, but users with limited conversational skills may have a negative experience. A more complex solution needs to be implemented in order for a user to feel comfortable talking to a chatbot.

## Future Work

The dating service developed in this study could be improved in a number of ways. For starters, a neural network trained on audio or video data will greatly enhance personality prediction accuracy (Yang & Glaser 2017). This type of input could be accommodated by reshaping the chatbot. Second, by utilizing a generative chatbot model trained on a large dataset, the chatbot's capabilities might be considerably enhanced, allowing for user conversations.

The quality and quantity of data collected from the user for personality analysis would most certainly improve as a result of this. Finally, two people's compatibility is more complicated than their personality similarities (Jarrett 2018). Other criteria, such as user interests and activity, should be taken into account when matching users to improve the algorithm's quality.

# References

Aldous, D. (2017). Elo Ratings and the Sports Model: A Neglected Topic in Applied Probability? *Statistical Science*, *32*(4). https://doi.org/10.1214/17-sts628

Donnellan, M., Conger, R. D., & Bryant, C. M. (2004). The Big Five and enduring marriages. *Journal of Research in Personality*, *38*(5), 481–504. https://doi.org/10.1016/j.jrp.2004.01.001

Iyengar, S. S., & Lepper, M. R. (2000). When choice is demotivating: Can one desire too much of a good thing? *Journal of Personality and Social Psychology*, *79*(6), 995–1006. https://doi.org/10.1037/0022-3514.79.6.995

He, X., Liao, L., Zhang, H., Nie, L., Hu, X., & Chua, T. S. (2017). Neural Collaborative Filtering. *Proceedings of the 26th International Conference on World Wide Web*. Published. https://doi.org/10.1145/3038912.3052569

Eharmony Editorial Team. (2021, September 7). *10 Online Dating Statistics You Should Know*. Eharmony. https://www.eharmony.com/online-dating-statistics/

Jarrett, C. (2018). *Is it better to be like your partner?* BBC Future. https://www.bbc.com/future/article/20181011-are-relationships-better-if-partners-are-more-similar

Laubu, C., Dechaume-Moncharmont, F. X., Motreuil, S., & Schweitzer, C. (2016). Mismatched partners that achieve postpairing behavioral similarity improve their reproductive success. *Science Advances*, *2*(3). https://doi.org/10.1126/sciadv.1501013

L. (2019, June 24). *Match launches UK’s first AI Dating Chatbot, Lara, on the Google Assistant*. UK Dating - Match. https://uk.match.com/p/dating-advice/match-launches-uks-first-ai-dating-chatbot-lara-on-the-google-assistant/

Mikolov, Tomas et al. “Efficient Estimation of Word Representations in Vector Space.” ICLR (2013).

N. Mustafa, A. O. Ibrahim, A. Ahmed and A. Abdullah, "Collaborative filtering: Techniques and applications," 2017 International Conference on Communication, Control, Computing and Electronics Engineering (ICCCCEE), 2017, pp. 1-6, doi: 10.1109/ICCCCEE.2017.7867668.

Pennebaker, J. W., & King, L. A. (1999). Linguistic styles: Language use as an individual difference. *Journal of Personality and Social Psychology*, *77*(6), 1296–1312. https://doi.org/10.1037/0022-3514.77.6.1296

Robins, R. W., Caspi, A., & Moffitt, T. E. (2000). Two personalities, one relationship: Both partners’ personality traits shape the quality of their relationship. *Journal of Personality and Social Psychology*, *79*(2), 251–259. https://doi.org/10.1037/0022-3514.79.2.251

*Gamification of love: a case study of Tinder in Oslo*. (2018, September 8). Duo.Uio.No. https://www.duo.uio.no/handle/10852/64406

Tiffany, K. (2019, March 18). *The Tinder algorithm, explained*. Vox. https://www.vox.com/2019/2/7/18210998/tinder-algorithm-swiping-tips-dating-app-science

*Powering TinderÂ® â” The Method Behind Our Matching*. (2019). Tinder Newsroom. https://www.tinderpressroom.com/powering-tinder-r-the-method-behind-our-matching

Wang, S., Kim, K., & Boerner, K. (2018). Personality similarity and marital quality among couples in later life. *Personal Relationships*, *25*(4), 565–580. https://doi.org/10.1111/pere.12260

Zaremba, W. (2014, September 8). *Recurrent Neural Network Regularization*. ArXiv.Org. https://arxiv.org/abs/1409.2329

Yarkoni, T. (2010). Personality in 100,000 Words: A large-scale analysis of personality and word use among bloggers. *Journal of Research in Personality*, *44*(3), 363–373. https://doi.org/10.1016/j.jrp.2010.04.001

Shi, G. Y., Kong, Y. X., Chen, B. L., Yuan, G. H., & Wu, R. J. (2018). Instability in Stable Marriage Problem: Matching Unequally Numbered Men and Women. *Complexity*, *2018*, 1–5. https://doi.org/10.1155/2018/7409397

*160 First Date Questions - The only list you’ll need.* (2021, September 27). Conversation Starters World. https://conversationstartersworld.com/first-date-questions/

# Appendix A App Component Structure

Diagram

Description automatically generated

# Appendix B Research Process Flow Chart

Diagram

Description automatically generated

Chart, diagram, box and whisker chart

Description automatically generated

# Appendix C Mobile User Interface

Graphical user interface, text, application, chat or text message

Description automatically generatedGraphical user interface, text, application, chat or text message

Description automatically generatedGraphical user interface, application

Description automatically generated

Graphical user interface, text, application

Description automatically generatedGraphical user interface, text, application, chat or text message

Description automatically generatedGraphical user interface, text, application

Description automatically generated

# Appendix D Example Dating Questions

1. What’s your most favourite place on Earth?
2. What small things brighten up your day when they happen?
3. What were some of the turning points in your life?
4. What habit do you wish you could start?
5. What’s the worst or best job you’ve had?
6. If you unexpectedly won Rs. 100,000, what would you spend it on?
7. What’s the hardest you’ve worked for something?
8. If you had the power to change one law, what law would you change?
9. What’s something you’re interested in that most people wouldn’t expect?
10. Where did you take family vacations to when you were younger?

# Appendix E Admin Dashboard

Table

Description automatically generated(concept)

# Appendix F User Survey Questionnaire

Text, letter

Description automatically generated

Text

Description automatically generated

Text, letter

Description automatically generated

# Appendix G Responses for User Survey Questionnaire

Chart, pie chart

Description automatically generated

Scatter chart

Description automatically generated

Chart, pie chart

Description automatically generated

Chart

Description automatically generated

Chart, pie chart, bubble chart

Description automatically generated

Chart, pie chart

Description automatically generated

Chart, pie chart

Description automatically generated

# Appendix H User Evaluation Questionnaire

Text

Description automatically generated

Text

Description automatically generatedGraphical user interface, text, application, letter, email

Description automatically generated

# Appendix I Gantt Chat

Timeline

Description automatically generated with low confidence