The Modern Day Cupid's Arrow

Mike Nelhams, Matthew Forster, Theo Mayor, Peter Smyth   GROUP 10

oa18502@bristol.ac.uk, zc18711@bristol.ac.uk, io18847@bristol.ac.uk ax18172@bristol.ac.uk,

February 22, 2021

Abstract

Online dating is a vast industry in the modern world and with hundreds of millions of people looking for love it is vital to narrow down the pool of potential partners. Throughout this report we look at two different wealthy sets of data regarding users of online dating websites and applications. The first data set is for understanding the qualities of people who use these sites and what they are all searching for in love. Through simple word analysis of given text responses, and comparisons of user attributes, we investigate deeper into which aspects users deem important.

The second dedication of this report is our unique algorithm for recommending users on dating sites. After researching into like-minded approaches, we strongly believed that creating a K-Nearest Neighbour (KNN) based recommender system would be able to create matches based purely on the ratings provided in the data. Demonstrating this successfully, and assessing the results, the algorithm we developed accurately recommends users, yet it possesses potential room for improvement if linguistic tools are also brought in order to quantify user information such as height, age or opinions. We provide deeper insights into the preferences of the users, although more of this type of analysis can be done to tailor the system to the users' demands. Expansion of our algorithm to incorporate the tools: 'LIWC' and 'XL-STAT', and enumerating word similarities into compatibility scores are the clear next stages of development. This would generate enhanced data for our current algorithm to utilise, when it assesses match strength.

1 Introduction

Using technology to find love is becoming increasingly prominent in the social media dependent world that we find ourselves living in. Online dating is seen as a safe way to find a partner by the majority of those under 50 [1] and it has become a multi-billion dollar industry [2], with millions of people frequenting sites such as Tinder, OkCupid and PlentyOfFish [2].

By first analysing the OkCupid data [3] we wanted to investigate into the type of user who be would typically be using dating sites. We found some further distributions, continuing the work carried out by the following report [1]. Then we began looking at creating our own recommendation algorithm for the ratings data [4], before finally looking at the weakness and the improvements that we could make, based on research on existing recommendation algorithms.

The implications of creating a successful algorithm would be vast, allowing customers to reduce the sizeable pool of potential partners to a smaller group. A lot of people know the struggles of online dating websites, where countless hours of swiping and messaging lead to matching with people with no similar interests and a dry conversation, where the relationship ends quickly. Confirming that both parties are looking for the same type of relationship and have shared interests would ensure that any potential connections would have a far higher chance of success (depending on the participants own definition of success). An important thing to note about the hypothetical development of our system, is that we would be able to constantly adapt and improve the algorithm as the client base increases (as is already done by many online dating sites [5]). Subsequently, this would help to increase the accuracy of the matches, and in the continuous process, it would improve the system by receiving useful feedback from users as to how successful their matches are. This is, however, beyond the scope of this project, but it is worth considering the ways in which our algorithm can be improved.

We aim to greater understand the methods behind matchmaking online, using two different sets of data. The second database looks at profile ratings [4], whereas the first set of data [3] contains more detail regarding user interests, hobbies and characteristics. Using these sets we will attempt to create an algorithm that will create possible recommendations to show to users as they use the site. This has two potential uses in actuality: either a service where details are collected and pairings are made based on the optimum collation of these details or a more streamlined service where there is some type of suitability measure, which the user can view to inform their decisions.

2 Statistical analysis

The first set of data contains a wealth of information regarding its users [3] and by performing different statistical analyses on the data we are able to comprehend what shared similarities there are between users and how important these similarities are to them respectively. All of the users are made anonymous in both data sets in order to protect their identities and the OkCupid data only represents one area of San Francisco. The set of profile ratings data [4] contains no personal information and therefore is considered irrelevant for statistical measures unlike the OkCupid data. All of code for the statistical analysis can be found in the following github repository: https://github.com/ax18172/group-10-R-code.

Some of the relevant features that we considered were the difference in ethnicity's within our data set and in San Francisco as a whole, and the comparison between the age of male users and female users. We were also very interested to investigate how a user's age affects: their views on astrology; their religious views; and their religious beliefs.

The first feature we analysed was the proportion of users who claim to be of a certain ethnicity. Figure 1 shows that the largest ethnic group using OKCupid, by far, was White. Many users claimed to be of more than one ethnicity, and as a result, our percentages do not sum to 100%, because some users are in multiple categories. We compare our percentages, shown in the figure 1, to a recent US census, [6]. The first observation is that the largest ethnicity 'White' is larger in our data set than in the city population, with the data having a 63%,[6], white population, compared to a 52.9% actual population. This difference is almost entirely accounted for by the under-representation in the Asian community. The data consisted of 13.6% of users with Asian heritage, compared to a 35.9% [6] representation in the actual community. Other minority groups within the city are more accurately represented within the data, such as Latino/Hispanic and Native American. We were unsure of the reasons for these differences and given more time this can be researched further.

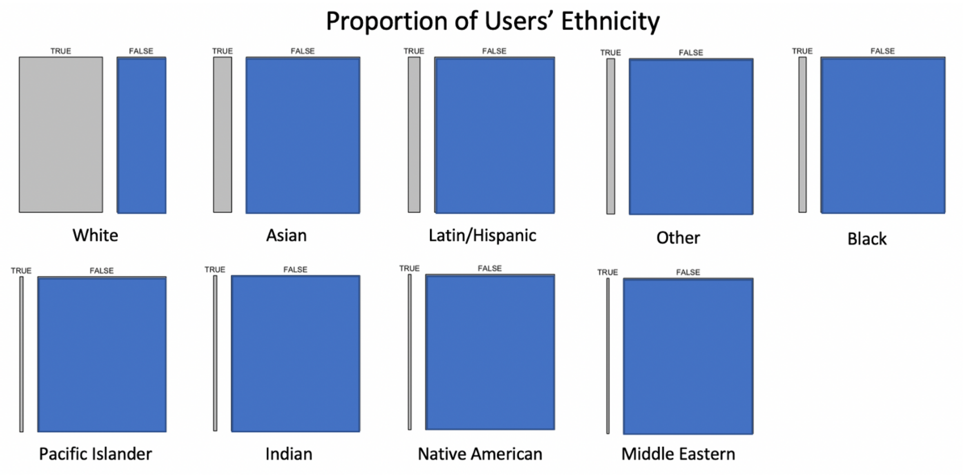

Figure 1: Representation of different ethnicity's within the data (%)

Regarding the correlation between astrology and age, 26% of the US population believe in the relevance of astrology, according to the Harris Poll [7]. Our data analysis, displayed in figure 2, shows that around 30% of all age groups believe astrology is 'fun to think about'. Is is also important that less than 2% of the participants believe that astrology 'matters a lot' to them, therefore it is a safe conclusion that astrology is not the driving factor for the majority of users of online dating.

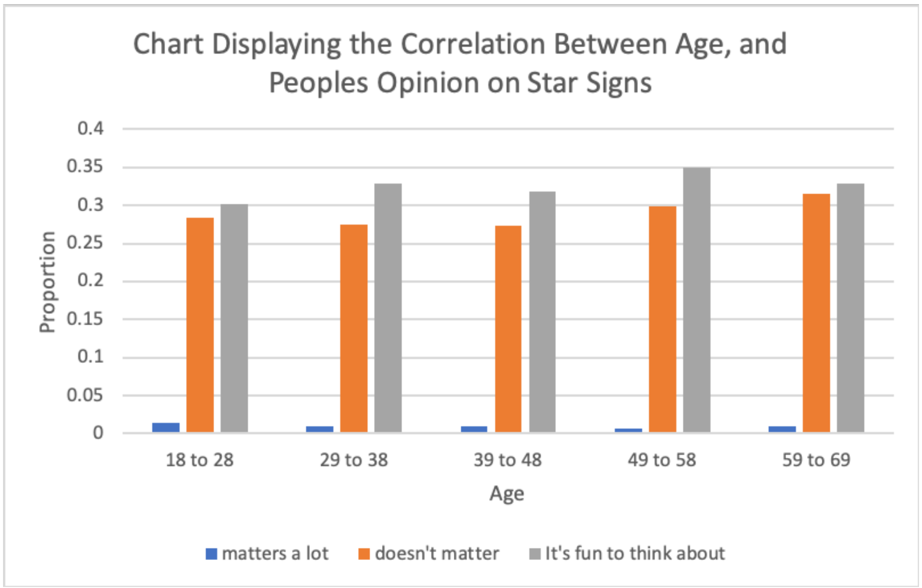

Figure 2: Chart representing how age and star sign are connected for the OkCupid data.

Another crucial factor in dating is age and we wanted to better understand the age distribution. The positively skewed normal distribution shows that the majority of the users ranged from between 29-40, which means that this site isn't popular with younger people, who may prefer other apps such as Tinder, Bumble, etc. This information can be improved with various filters, so we can be more decisive about the age of older users (looking for older users), as there are potential less matches. The actual distribution is affected by fake accounts, for example the second graph extends to the age 110, although we have no proof, we doubt this is a legitimate account. More information would need to be collected on the users' preferred age range for matches, in order to successfully use this as a compatibility measure.

The next feature that we analysed was the proportion of users in the age groups: 18 to 28, 29 to 38, 39 to 48, 49 to 58 and 59 to 69. The number of users in each age group is not consistent, figures 4 and 5 display the proportion of people in that age group, rather than the number of people. Before we analysed the data, based on our surroundings, we hypothesised that Atheism and Agnosticism would be more popular amongst users of younger ages. Although the majority of users aged 18 to 48 made no mention of religion in their profiles, figure 4 shows that our prediction was correct for the users that did specify their beliefs. Figure 4 shows that in most cases, a user's age is correlated to their religious beliefs. It shows that as age increases: the number of people who follow Agnosticism and Atheism both decrease; the number of people who claim to follow Buddhism and Judaism both increase; the number of people who follow Christianity and Catholicism both stay roughly consistent.

Where figure 4 gives information about what people believe in, figure 5 shows how devoted to their respective faiths the users are. In conjunction with figure 4, the figure 5 shows that the majority of users aged 18 to 48 made no mention of religion. This is interesting, because even though it shows no indication of whether the user is religious or not, it implies that the majority of people do not consider religion as an important factor for their relationship. Even from of the users who did specify their religion, across all of the age groups, the majority stated that they are 'not too serious' about religion. For all age groups, the least number of people indicated that they were very serious about the religion they follow, however as age increased, the proportion of people who were very serious about religion also increased.

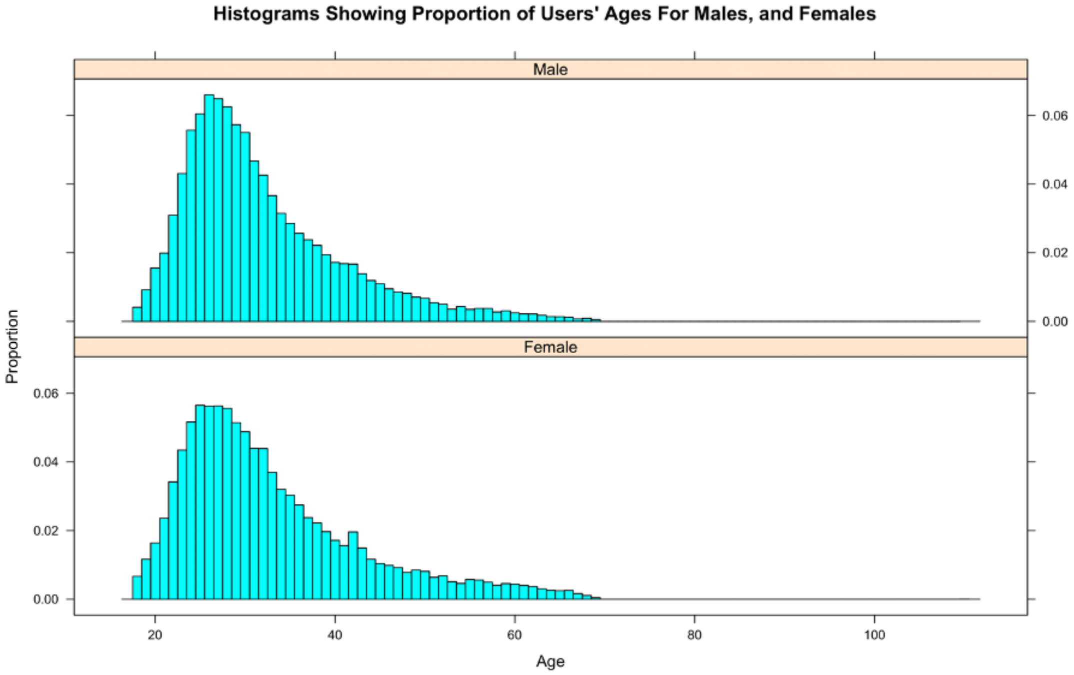

Figure 3: Positively skewed Normal distribution of ages within the data population.

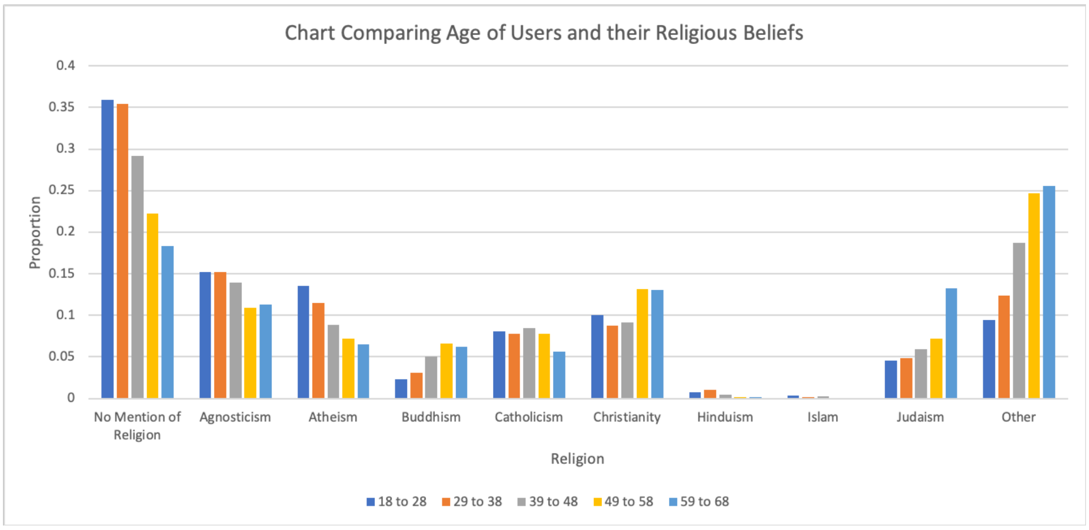

Figure 4: Chart displaying different age groups' religious beliefs

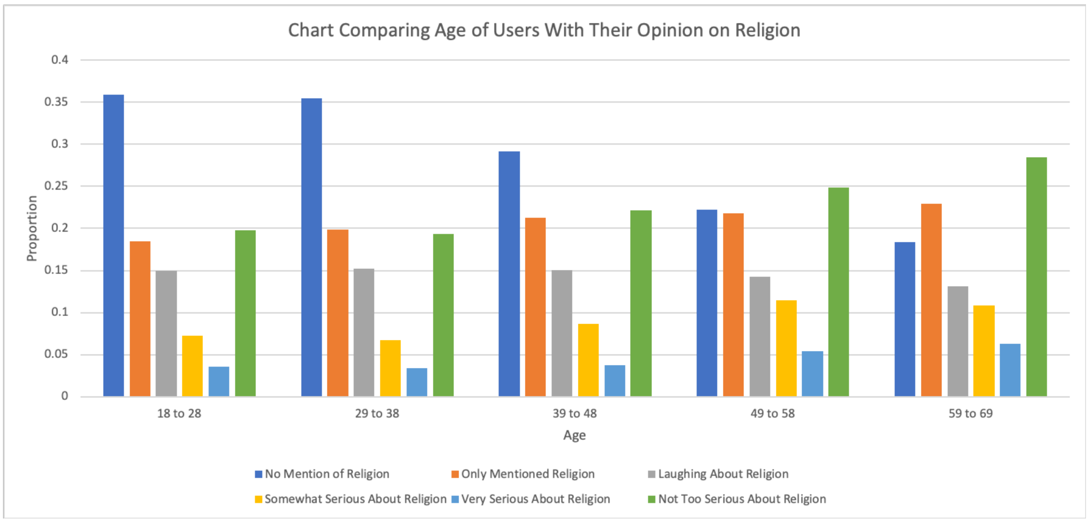

Figure 5: Chart displaying different age groups' opinions on religion

The primary question that the users were asked by OkCupid was 'what are [their] interests'. The word frequency in their responses, with the common stopping words such as 'I', 'it' and 'thing' were all removed, leaving relevant nouns, verbs and adjectives are all displaying in a word cloud chart in figure 6. From observing the wordcloud it can be inferred that work is the most important interest for the demographic, since the words 'work', 'working' and 'job' are some of the largest in the cloud. 'San Francisco' and 'bay area' are also often mentioned, due to the location of the users and not relevant to the population at large, highlighting how important to recognise this data is specific and not generalised. The wordcloud was also created to show a simple ability of word recognition. If the analysis is continued, it is possible to look into numerating matching words or phrases for compatibility analysis.

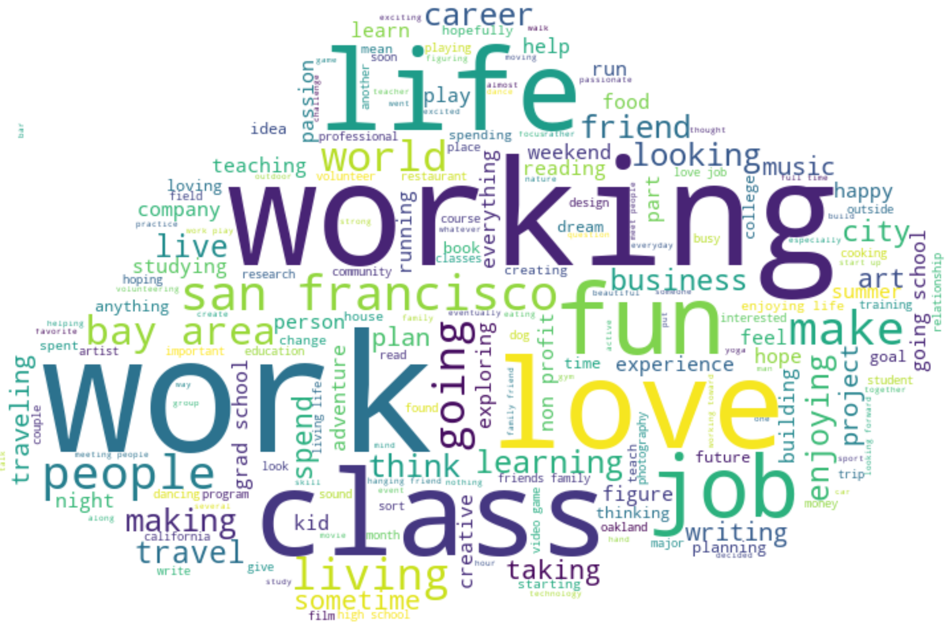

Figure 6: Wordcloud representing the frequency of words in the responses of users to the question: 'what are your interests?' Larger words correspond to a greater frequency.

3 State of the art pairing methods

As one of our data sets was provided from users of the site OkCupid, we researched existing algorithms OkCupid used. We also researched the system Tinder uses as the profile rating data given to us is presented in a similar way to this website - no profile data or interests were provided, just a rank between 1 and 10 of attractiveness and a gender for each user in the data set.
Firstly, in Tinder, users are shown the profiles of other users in rapid succession and prompted to swipe right on their phone to indicate that they'd like to begin a conversation with that person, or to swipe left if they are not interested. If two people swipe right on each others profiles, then a conversation can begin. Tinder varies which profiles can swipe on which by using a dynamic score ranking system (similarly to the ELO system in Chess [8]) to rank all of the users based on their "desirabilities" [9] - this indicates how likely another person is to swipe right on a profile. The users' desirabilities are therefore affected not just by attractiveness but also on profile information such as age, distance etc. If a user is swiped right on by someone with a higher score, their own score increases and vice versa. Users are provided the opportunity to swipe on people with similar scores via the Tinder recommender system.

OkCupid works by presenting users with many personal questions when they initially join - for each question they select their answer and they select the answers that they would prefer a partner to pick and how important that question is to them. A matching algorithm is then used on this data to find a compatibility percentage, between 0 and 100%, between the user and potential partners. The compatibility percentage is later displayed to both parties. Unlike on Tinder, users can message any potential matches, although those with high compatibility percentages are shown first. Compatibility is based on answers to questions, thus OkCupid's system relies more on similarities in character than physical appearance in contrast to Tinder, because OkCupid present themselves as a site for finding longer-lasting, more serious relationships than Tinder.

Implementing a ranking system like Tinder's requires constant user input to generate and change the rankings of users as time progresses. OkCupid's system, however, requires a substantial amount of information on each user to find suitable potential matches. We decided to use the data from profile ratings [4] in our approach and therefore developed a system more similar to Tinder than OkCupid, due to the fact that the profile ratings data lacks all personal information regarding the users, so a user similarity based system would be seemingly implausible unless the data was somehow quantified and assigned respective ratings.

4 Research into reciprocal recommendation systems

We wanted to consider quantifying the OkCupid data, in order to create a numerical estimation of the compatibility between the members. However, this proves itself to be a difficult problem to solve, with a great deal of the data containing subjective answers, which would be pragmatically difficult to quantify, furthermore with each user differing in their style of language- in addition to the grand scale of the data. We felt it crucial to look into the existing methods which have attempted to achieve this, so we could evaluate the possibility of quantifying the OKcupid data.

Initially, the paper [10] uses the text analysis program 'Linguistic Inquiry word count' (LIWC) to analyse the essay questions and gain insights to users. The LIWC program splits up words into 80 categories, outputting percentages of each category for further analysis. By looking at a portion of these categories, which potentially contained gender biases and trends, conclusions can be drawn as to optimising the profiles for maximum success and what features could presumably be 'desirable' to the majority. Although this paper does not quantify the strength of the matches, but instead clusters language similarities between the genders, it does show the possibilities of using the analytics tools of LIWC and XLSTAT in order to quantify the user data.

We aimed to be able to generate a numeric value for the strength of matches. From research, the best way to do this would be a reciprocal recommendation system. The benefits of this system over a typical recommendation system are highlighted in [11]. Success depends on both parties involved in the comparison, not solely the party to whom you are recommending to, but also the recommended party. This secures that both members dynamically receive the optimum match, without the system biasing any particular users repeatedly. In order to create numerical values to recommendations and matches, this paper uses responses much similar to those in [3], the 'Recon algorithm' [11] measures the extent to which the questionnaire responses concur, and then it outputs a score, $0 \leq$ Compatibility rating ≤ 1, with one being the perfect match and 0 representing the worst possible match. With the RECON algorithm, distributions are generated from messaged transmitted by the user and each distribution gives different useful information such as which body types, education and relationship are preferential to an active user. The preference of a user x and their preference, P_a can be shown as a list of these distributions, such that.

$$P_x = P_{x,y} : \forall y \in Y[11] \tag{1}$$

Using the generated preferences, they were then able to work out a 'compatibility score'. The Recon method is similar to the one we would have liked to have used, an algorithm which improves over time. This is due to the fact that more accurate distributions, (gained from increased activity),lead to a deeper understanding of each user's preferences. Although, it still faces the same cold-start problem, such that a new user has no known distributions, this particular paper addresses this purely by suggesting that established users are to be frequently matched with new users. As the new users are being recommended so frequently, this evidently means that the new user's preference distributions are built up quickly, so that the system can start producing more apt estimations regarding mutual connectivity.

The next research paper looked at, [12], further develops the aforementioned RECON algorithm. The first adaptation of RECON was to force continuity on previously grouped ages, such that similar ages were rewarded but variation in age was not a deciding factor. As improvements to their algorithm are made, more similarity measures are taken into consideration. These three measures are content-similarity (2), interest similarity (3) and attractiveness similarity (4). These similarities were defined in [12], as:

$$content - similarity(x, y) = \frac{\sum_{a \in A_x \cap A_y} P_a(x, y)}{|A_x \cap A_y|}, \tag{2}$$

$$interest - similarity(x, y) = \frac{|Se(x) \cap Se(y)|}{|Se(x) \cup Se(y)|}, \tag{3}$$

$$attractiveness - similarity(x, y) = \frac{|Re(x) \cap Re(y)|}{|Re(x) \cup Re(y)|}. \tag{4}$$

Content-similarity shares resemblance to the RECON method, looking for common supporting attributes and respectively associating a numerical value. A_x and A_y are the lists of attributes for users x and y, and P_a is the calculated value of these preference attributes to the other user. Interest-similarity and attractiveness-similarity are focused on the interactions between the users. $Se(x)$ is the set of all 'out-neighbours', i.e. the set of the messages that x has sent, 'its cardinality reflects the activeness of x' [12]. By promoting those users who have messaged the same users successfully, we could look at a common-interest for users and therefore we could recommend similar people. Lastly, the attractiveness-similarity, calculated in much the same way as interest-similarity, alternatively measures almost the opposite. $Re(x)$ is the set of 'in-neighbours', so the amount of messages this user receives, the cardinality of $Re(x)$ can be described as the attractiveness of x.

Although not accurate for new users (cold-start problem), the Recon process improves in time with more messages as it builds up a more details profile of your attractiveness and interest. These examples of previous research show a clear possibility for the success of these algorithms, however constant updates and adaptations would required in order to refine it further, perhaps additionally including the alternate personality measures. A more straightforward and accurate approach can be implemented with the second set of data [4] and this is the main goal for the report.

5 Recommender system for user-user rating data

5.1 Notation

The notation for the recommender system in this report is displayed in figure 7. Note that the rating prediction $P_{a,j}$ is different to the notation for user preference $P_a(x, y)$ in the aforementioned research papers [11] and [12].

Variable	Description
a	The active user using the dating system
j	The target user for a to rate
n_i	The i-th Nearest Neighbour to the active user a
ζ	A normalising factor
$\overline{r_a}$	The mean rating **for** the active user a
$\overline{r_j}$	The mean rating **of** the target user j
$\overline{r_{n_i}}$	The mean rating **of** the i-Nearest-Neighbour n_i
$\mathbf{r_{x,y}}$	The rating vector that any user x has given **for** any user y
$P_{a,j}$	The prediction rating which user a gives user j
N	The total number of users within the system
k	The maximum number of K-Nearest Neighbours used for predictions
$\overline{r_t}$	The mean rating of every user within the system
B	The bias factor for user popularity

Figure 7: Index of notation

5.2 Method

5.2.1 Choosing an appropriate algorithm

A user-user recommender system is an important system that recommends to an active user a new profiles for them to rate on an online dating site. It's important to consider that there exist numerous recommender systems, due to the increased popularity of dating sites and apps in the modern era, because each company wishes to claim their algorithm is the best. User rating data is abundant in data science and it is also crucial to recognise that each specific case of rating data has best-suited algorithms. In the case of assigning each user a respective suggested user, user-item algorithms that require a utility matrix, such as SVD [13], are severely poor choices. Recommender systems that require utility matrices are poor choices, because if a matrix is constructed for N users, then the utility matrix will be an NxN matrix. The space complexity will therefore be $O(N^2)$, which is reasonable for a small number of users, such as 1,000. However for the ratings data set we use [4], the number of total users is 220948, which means this can be represented as a minimum of 48,818,018,704 bits or rather approximately 6 gigabytes. A matrix of this size (220948 by 220948) is highly impractical to manipulate and would only become worse as the dating system gains new users. The distinction between item-item, user-item and user-user systems is explored further in detail in [14].

The method proposed by this project is a user-user system that predicts how an active user a would rank a target user j based on the given data of every user. The idea is that we can predict how the active user a would rate some user j, both based on how a rated other people and based on how users similar to a rated the target user j. All of the recommender system and data used is included in the following github repository: https://github.com/MikeMNelhams/Recommender-System-for-User-User-Ratings.

The next most important feature for this particular recommender system is the similarity measure. The users' suggestions will be based on their similarity to other users: who they each rated. There exist various measures for comparing multidimensional data points and the effectiveness of all these measures [15] will be evaluated in order to decide which is the most apt for this set of ratings. The proposed similarity measure for this report is Pearson's product-moment correlation (PMCC), because the output is normalised between -1 and 1 which is vital for providing rating data that is later normalised between 0 and 10. Other similarity measures such as Manhattan or euclidean, which are both distance measures, are ineffective at measuring the correlation between the K-Nearest Neighbours' (KNN) ratings, due to the lack of normalised outputs, therefore resulting in ratings exceeding 10. For finding the KNN, distance measures are more appropriate, with the most efficient directionless distance measure being the euclidean distance measure [16].

The standard formula used for dating recommender systems that utilise the KNN algorithm [17] is:

$$P_{a,j} = \overline{r_a} + \zeta \sum_{i=1}^{k} w(a, n_i)(r_{n_i,j} - \overline{r_j}) \tag{5}$$

where n_i is the i-th nearest neighbour to the active user a. ζ is a normalising factor and $w(a, n_i)$ is the PMCC [17] [15] between a and j, with i summed over the users which both a and j have rated.

$$w(a, n_i) = \frac{\sum_i (r_{a,i} - \overline{r_a})(r_{j,i} - \overline{r_j})}{\sqrt{(\sum_i (r_{a,i} - \overline{r_a})^2 \sum_i (r_{j,i} - \overline{r_j})^2)}}. \tag{6}$$

5.2.2 Normalizing the predicted ratings

An issue with equation (5) is that the predicted ratings range from:

$$-10k\zeta \leq P_{a,j} \leq 10\zeta(k+1). \tag{7}$$

Since k is the number of the KNN used in the calculation and is unbounded, the predicted ratings do not have well-defined limits. Setting the normalization factor to the following:

$$\zeta = \frac{1}{k} \tag{8}$$

changes the range of the predicted ratings in (7) more simply to:

$$-10 \leq P_{a,j} \leq 20. \tag{9}$$

For the purposes of comparing which users should be recommended for the active user first, the output predicted ratings between -10 and 20 are sufficient, however these predicted ratings do not accurately mirror the users actual ratings, since the user is limited to rating between $0 \leq P_{a,j} \leq 10$. To map the domain of predicted ratings to the normalized range 0 and 10, whilst still conserving the ranking of the predictions, a sigmoid function [18] can be used.

The bijective sigmoid function (10) shown in figure 8 continuously asymptotes to $y = 0$ and $y = 10$, which normalizes the output to the range: $0 \leq \sigma(x) \leq 10$.

$$\sigma(x) = \frac{10}{1 + e^{-\frac{x}{2}}}. \tag{10}$$

5.2.3 Biasing unrated users

The prediction equation (5) is effective at extrapolating how a rates j, but only in the specific cases where there is sufficient data. Therefore if the relevant data exists in the database, it is expected that:

$$\sigma(P_{a,j}) \approx r_{a,j}. \tag{11}$$

Subsequently, by calculating the predicted ratings $P_{a,j}$ across all the users j within the database, then sorting the predictions from greatest to smallest and removing the users that a has already

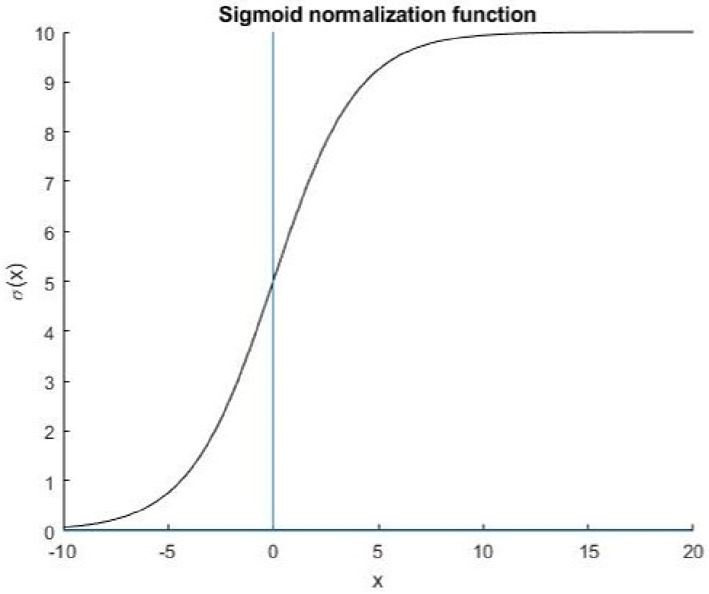

Figure 8: Sigmoid transform function.

rated, the recommender system can effectively suggest to the active user a new users for them to rate. However the significant condition that the sufficient data exists is not always true for every user. The database is necessarily sparse, meaning that not every user has rated every other user and this is expected from every standard dating database. For a data set with N users, the number of ratings for a complete data set is $N^2 - N$, assuming that users cannot rate themselves. For the following data set, there are 220,948 existing users. As illustrated in figure 9 it is very impractical and near impossible to have a complete data set, because there would have to be almost 50 billion total ratings, an average of 226,300 ratings per user. There are 17,359,345 total ratings in the data set, which is clearly negligible compared to the required 50 billion.

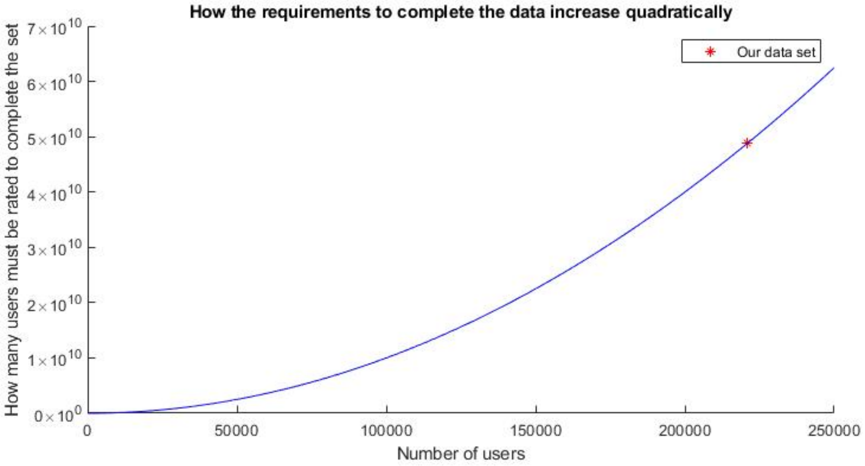

Figure 9: The number of ratings required to complete the data increases quadratically with the number of users.

Amongst online dating sites there are numerous users, each with their own varying levels of scrutiny in their rating. Additionally, every data set of users is certain to have users who have not been rated by other yet. This may be due to the fact they have recently joined the dating site, or that they have never been shown to others. An important distinction for them is determining their mean rating, this is the cold-start problem. If nobody has rated them, then the calculated mean rating for them is effectively 0, but this is unjustifiable, since this will therefore disadvantage new or unpopular users. Disadvantaging new users causes the dating system to become stale, making the site less enticing to returning users. To avoid this, the default mean for an unrated user is the mean rating of every user on the dating site. For the given database, the mean rating value (r_t) is 5.938 accurate to 3 decimal places.

In the case in which there is incomplete data, rather than skipping the unrated users like performed in the report [17], we propose using the mean rating value for the data $\overline{r_t}$, and a bias factor B that is inversely proportional to the number of ratings that a user has. This implies that the less popular users are more likely to become recommended and similarly the more popular users are less likely to become recommended. This changes the formula in (5) to become:

$$P_{a,j} = \overline{r_a} + \zeta \sum_{i=1}^{k} w(a, n_i)(r_{n_i,j} - \overline{r_j}) + B \tag{12}$$

where the bias factor B is:

$$B = \frac{\overline{r_t}}{1 + |\mathbf{r_j}|} \tag{13}$$

and if $|r_j| = 0$ then $\overline{r_j} = \overline{r_t}$. The bias has the effect of proportionally balancing out the dating system, creating an immediate and exciting response for new users, because they are more likely to become rated and it does not make the popular users exponentially more popular, which can also be considered to be possibly unfair to new or returning users. Additionally, with the sigmoid function (10), despite including the bias, the predicted rating range remains $0 \leq \sigma(x) \leq 10$.

5.3 Results

5.3.1 Evaluating the bias effectiveness

Comparing the predicted ratings data for $k = 10$ and an example active user of $a = 1$, the data with and without biases can be seen in figures 10 and 11. The smallest number of ratings for the data is 20 ratings, corresponding to an maximum increase of approximately 0.35. The figure clearly displays that the increase in prediction is inversely proportional to the number of ratings that each user has, which is the desired feature of biasing. The difference in data spread can be seen more clearly in figure 12. The interquartile range of the data increases drastically, whilst the minimum, maximum and median are all changed negligibly. The decrease in data density is advantageous since this allows each of the different predictions to be more easily distinguished from each other.

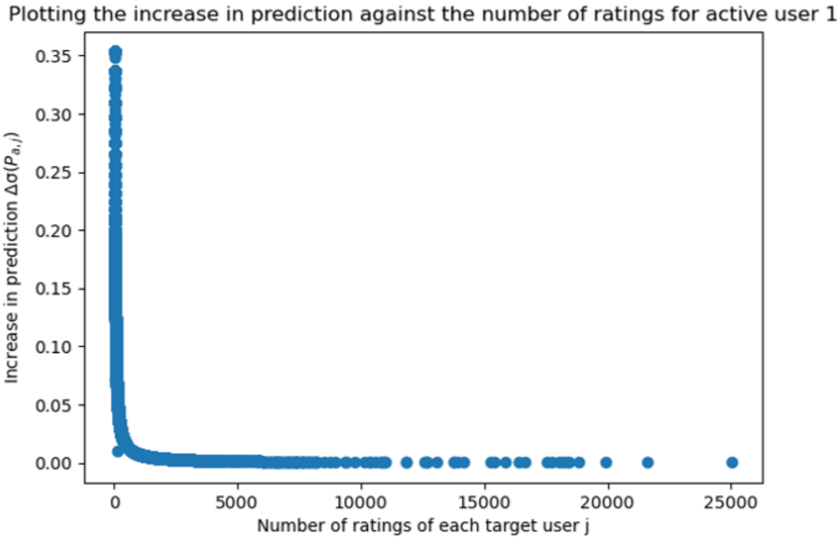

Figure 10: How the biasing affects the prediction results individually.

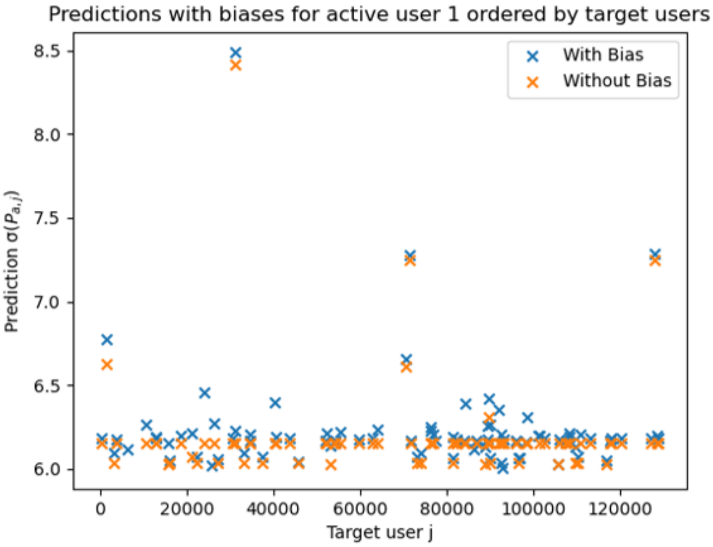

Figure 11: The effect of biasing on the predicted ratings. (Predicted ratings less than 6 are omitted to make the graph less dense and more readable).

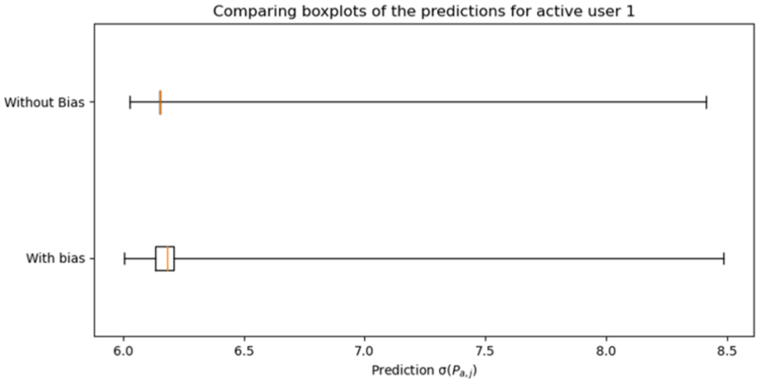

Figure 12: The effect of biasing on the predicted ratings and how the biasing affects the spread of data.

5.3.2 Time complexity of calculation and accuracy of the predicted ratings

The maximum number of KNN (k) used for evaluating predicted ratings has two effects on the recommender system. Firstly, as k increases the computational run-time of calculating the predictions increases linearly, as shown in figure 13. Interpreting the gradient of the graph, it takes nearly three more seconds per additional nearest neighbour involved in the calculation. Secondly, as k increases, more information is being taken into account to calculate the predicted ratings and therefore the predictions become more accurate. However, the PMCC (6) decreases if the i-th nearest neighbour is less similar to a which means that the effect of k increasing becomes more negligible and the predicted ratings will eventually converge to fixed values as shown in figure 14. Selecting a value of $15 \leq k \leq 25$ calculates the predicted ratings, in fewer than 140 seconds for our code and is sufficiently large to be considered an accurate prediction.

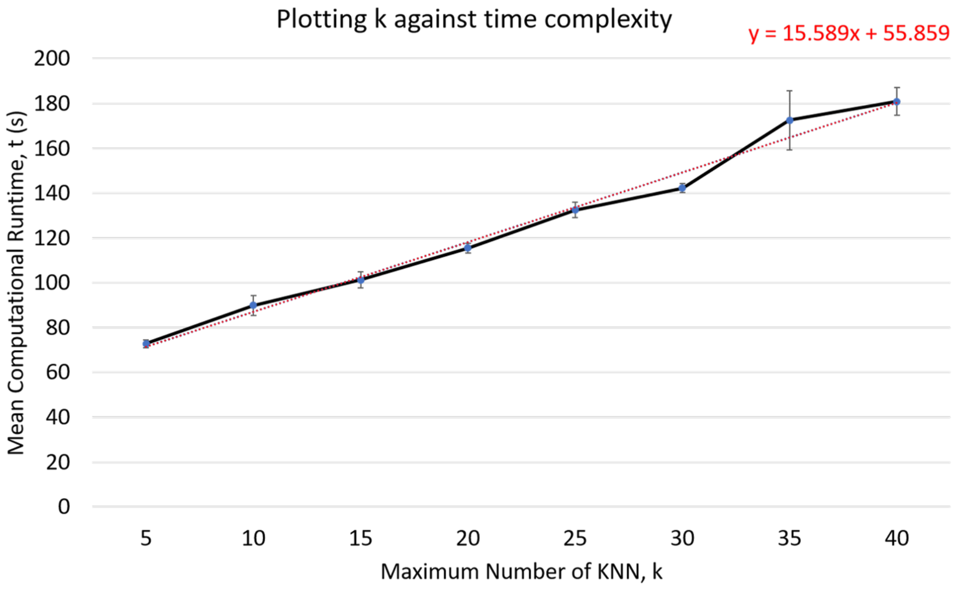

Figure 13: The graph shows the effect of how increasing k increases the computational run-time (s). Data is gathered from the active user $a = 1$.

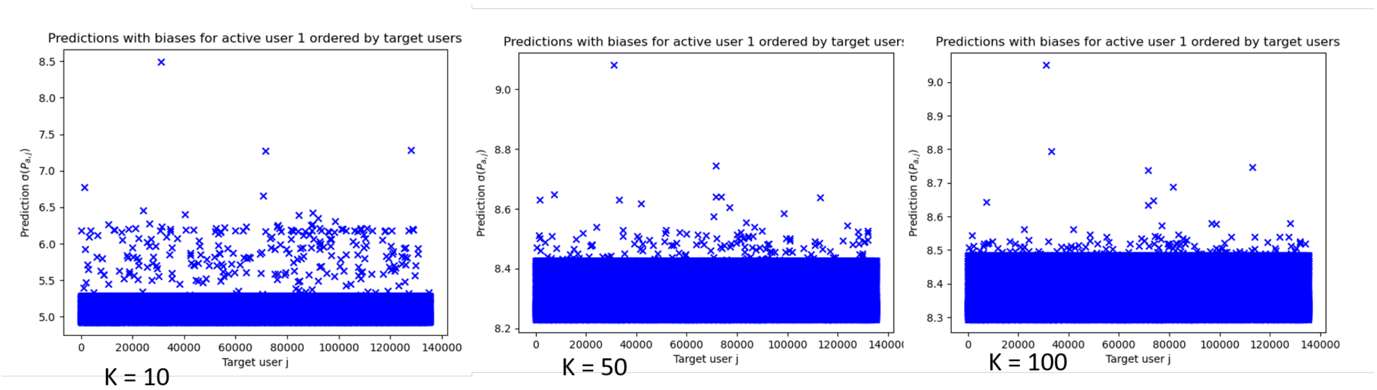

Figure 14: The 3 sub-figures show how increasing k increases the accuracy in predicted ratings. The data points show less change as k increases, with the differences between $k = 50$ and $k = 100$ being difficult to spot. Data is gathered from an active user $a = 1$.

6 Further improvements and discussion

Our approach for a recommender system differs from OkCupid's in the sense that ours requires users to provide a rating for other users before generating the predictions - OkCupid instead has users answer questions. Although, our approach is more similar to Tinder's, in that it requires user input to generate a list of predicted ratings [9]. Unlike Tinder, our approach does not generate a ranking list of people's 'desirability' and our system does not recommend people of similar rankings with one another - instead it looks for people who've been rated similarly by potential partners and matches the user with potential partners who have rated this similar user positively. This system ensures that users are given reasonable and accurate partner recommendations - however similarity or popularity base systems guarantee shared interests between potential matches.

If given more time, we desire to further our system by using programs such as LIWC and XL-STAT to incorporate further word analysis into our predictions. This would have increased accuracy by measuring various types of similarity between users, and then actively developing the system with user input. With such huge implications in narrowing down the significantly large dating pool. A refined algorithm would assign distributions to multiple aspects of users' interests, and it would successfully ensure that both users have shared interests in each other. The greatest quality about an algorithm such as this, is that the more interactive the users, the more accurate the matches will become, which equates to continuous improvement. A slight weakness of our initial system is the substantial computational run-time (s) with higher k values. Improving this by running the algorithm for individual users (preferably using their device), the running time can be kept minimal, therefore it would be an improvement on the current running of our algorithm.

A final improvement, perhaps unconsidered throughout this report, is the limitations of our data itself, not only the heterogeneity of the data, but also the fact that it may not be completely representative of society. In theory, a comprehensive algorithm such as that in [12], would be able to understand the users sexual preference and ensure that the matches are successful. Therefore if a separate data set was given with homogeneous data, the algorithm would perform the same operations and could transition between data sets fluidly. Since our data set and the one used in the algorithm [12] are heterogeneous, we cannot be absolutely certain of the prosperity of these algorithms. As a result, more research and more testing would need to be done on homogeneous data in order to evaluate what aspects change, if any, and what improvements should be made to the algorithm to sample heterogeneous data.

In conclusion, we provide useful insights to the data we were presented with [3] in terms of the statistical analysis; the creation of word clouds and simple distributions allowed us to more clearly understand the similarities and preferences of the users. Notwithstanding the analysis, the particularly exciting investigation for Cupid's Arrow was our ability to obtain accurate rating predictions out of a large amount of ratings data [4]. With more research and development of our algorithm and including previous state of the art research, great improvements can be made to ensure: that the recommendation algorithm confidently provides accurate recommendations, which take into account similarities, preference and the users' ratings and that the system can quickly narrow down the pool- from thousands of people to but a few suitable candidates.

7 Bibliography

References

[1] *Research on American's feelings on online dating* [online]. Available from: https://www.pewresearch.org/internet/2020/02/06/americans-opinions-about-the-online-dating-environment/, 2020 - February. [Accessed 25 April 2020].

[2] *Match Group Investor Relations website* [online]. Available from: https://ir.mtch.com/overview/default.aspx, 2020 - April. [Accessed 20 April 2020].

[3] Everett Wetchler. *OKCUPID Anonymous Data (California)* [online]. Available from: https://github.com/wetchler/okcupid, 2016 - May. [Accessed 5 April 2020].

[4] Vaclav Petricek. *Collaborative filtering dataset - dating agency* [online]. Available from: http://www.occamslab.com/petricek/data/, 2006 - April. [Accessed 7 April 2020].

[5] *OkCupid Blog Post: "We Experiment on Human Beings"* [online]. Available from: https://web.archive.org/web/20170216190818/https://theblog.okcupid.com/we-experiment-on-human-beings-5dd9fe280cd5, 2014 - July. [Accessed 2 May 2020].

[6] United States Census Bureau. *U.S. Census* [online]. Available from: https://www.census.gov/quickfacts/sanfranciscocountycalifornia, July 1, 2019. [Accessed 7 May 2020].

[7] The Stagwell company. *The Harris Pole* [online]. Available from: https://theharrispoll.com/new-york-n-y-december-16-2013-a-new-harris-poll-finds-that-while-a-strong-majority-7
December 16, 2013. [Accessed 6 May 2020].

[8] *Elo system in Chess* [online]. Available from: https://en.wikipedia.org/wiki/Elo_rating_system, 2020 - April. [Accessed 27 April 2020].

[9] *Fast Company article on the Tinder algorithm* [online]. Available from: https://www.fastcompany.com/3054871/whats-your-tinder-score-inside-the-apps-internal-ranking-system, 2016 - November. [Accessed 27 April 2020].

[10] Meenakshi Nagarajan and Marti A Hearst. An examination of language use in online dating profiles. In *Third International AAAI Conference on Weblogs and Social Media*, 2009.

[11] Luiz Pizzato, Tomek Rej, Thomas Chung, Irena Koprinska, and Judy Kay. Recon: a reciprocal recommender for online dating. In *Proceedings of the fourth ACM conference on Recommender systems*, pages 207–214, 2010.

[12] Peng Xia, Benyuan Liu, Yizhou Sun, and Cindy Chen. Reciprocal recommendation system for online dating. In *2015 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*, pages 234–241. IEEE, 2015.

[13] Mayukh Bhattacharyya. *Beginner's Guide to Creating the SVD Recommender System* [online]. Available from: https://towardsdatascience.com/beginners-guide-to-creating-an-svd-recommender-system-1fd7326d1f65, 2019 - November. [Accessed 17 April 2020].

[14] Baptiste Rocca. *Introduction to recommender systems* [online]. Available from: https://towardsdatascience.com/introduction-to-recommender-systems-6c66cf15ada, 2019 - June. [Accessed 17 April 2020].

[15] Marvin Lüthe. *Calculate Similarity — the most relevant Metrics in a Nutshell* [online]. Available from: https://towardsdatascience.com/calculate-similarity-the-most-relevant-metrics-in-a-nutshell-9a43564f533e, 2019 - November. [Accessed 17 April 2020].

[16] Saikat Bhattacharya. *A short introduction to distance measures in Machine Learning* [online]. Available from: https://towardsdatascience.com/a-short-introduction-to-distance-measures-in-machine-learning-886fb579d148, 2019 - February. [Accessed 4 May 2020].

[17] Lukas Brozovsky and Vaclav Petricek. Recommender system for online dating service. Available from: http://www.occamslab.com/petricek/papers/dating/brozovsky07recommender.pdf, 2007. [Accessed 1 May 2020].

[18] Florian Bansac. *Sigmoid function* [online]. Available from: https://ailephant.com/glossary/sigmoid-function/, 2018 - January. [Accessed 5 May 2020].