Modern Matchmaking:

Utilizing artificial intelligent techniques to modernize the traditional process of matchmaking.

Agathe Benichou and Michael Teddick

CS420: Artificial Intelligence

Lafayette College

December 7th, 2017

Submitted to Professor Chun Wai Liew

Table of Contents

A. Abstract Summary

After the completion of three group projects whose problem statement was given by Professor Liew, the culmination of the course was an individual project whose problem statement was chosen, rather than given. Each individual was tasked with creating a problem that could be examined through fundamental artificial intelligent algorithms and techniques. This proposal, in addition to the presentation, will serve as an exploration as to how the chosen problem at hand can potentially be solved. Our chosen problem is to provide traditional matchmakers with an artificial intelligent tool to aid in their process of matching individuals in their community for marriage. By utilizing web scraping, relational models, profiling techniques and learning algorithms, our tool will increase the number of successful matches made by the matchmaker, while still retaining the time-honored skills and preferences of a matchmaker.

B. Introduction

Matchmaking is the process of matching two individuals together for the purpose of marriage. Whether it is done by a Hindu astrologer, a Jewish shadchan or a matriarch in your community, a matchmaker can be an essential tool in creating successful and happy unions. According to Merriam-Webster, a matchmaker is someone who tries to bring two unmarried individuals together to promote a marriage. A matchmaker is typically a member of the community that they are setting up matches for. Despite having the old-world reputation as depicted in the 1971 film *Fiddler On The Roof* by Norman Jewison, it is an attractive option for modern individuals coming from a variety of cultural and religious backgrounds. This process is quite different than that of an online dating website, such as eHarmony.com or JDate.com, as it involves personal recommendations and careful scrutiny by all parties involved. While online dating websites or mobile applications are popular mechanisms for the youth of recent generations to find love, it is not the preferred way of those living within orthodox religious and cultural communities. In addition, the modern lifestyle and the international dispersion of close-knit communities has aided this ancient tradition in gaining reputability as it offers an individualized search with careful screening, confidentiality and dating advice.

C. Problem Statement

In today's world, successful and modern matchmakers are being swamped with very eligible candidates within their communities looking for marriage. As a result, they are having trouble keeping up with the demand using their traditional methods. They are looking for a revolutionary recommender system that is easily downloadable onto their personal computers which they can use to ease their work while still creating prosperous matches for their communities. They want to input the client's personal and family information, as well as dating preferences, into the system to create a profile for that candidate. The system should sift through potential matches already in the system to return the client's best match. However, the matchmakers still want to retain their conventional and personal methods of matchmaking, such as meeting with their clients in order to get to know them prior to matching and meeting with their clients for feedback on their suggested matches, while utilizing artificial intelligent techniques to enhance the process.

D. How it works?

Matchmakers from around the globe will be able to download the program onto their personal computers. Once downloaded, they can initialize the program with their matching preferences. For example, one matchmaker might believe that familial similarity is the most important criteria for a successful match, while another may believe that similar interests or personality types are of top importance. A matchmaker will be able to express and retain their matching preferences through this program initialization. These preferences will be utilized in later stages of matching. After this initialization phase, the program is ready to help create matches.

An individual seeking a matchmaking service will fill out an extensive form about themselves which includes personal information, lifestyle, family information, professional ambitions and dating preferences. This generic form will be given to all the clients and will mostly compose of multiple choice questions. Once the client has completed the form, they will send it to the matchmaker so that they can understand what kind of person they are. The matchmaker meets with the client after they have read over their form as an opportunity for the matchmaker to familiarize themselves with their client and to identify character traits or particularities not indicated in their form. This allows the matchmaker to ask any clarifying questions about their form. This is also a chance for the client to express any desired qualifications for a possible significant other such as age ranges and religious constraints. After the meeting, the matchmaker will load the clients form and their own observations into the system, thus creating a profile for the client within the system. The system will use this explicit data to scrape the web about the client in order to obtain additional information about the client. Any newly collected implicit input is added to the clients profile and now the profile is ready for matching. This profile will be continuously updated throughout the matchmaking process.

Once the user has a profile comprised of both explicit and implicit information, they are ready to be run through the system. The system will utilize the strict requirements indicated in the user's profile to formulate a pool of eligible matches for that user. During this step, the system rules out any candidates that the client has specifically requested to be ruled out. From this pool of eligible candidates, matches are based on preferences and are ranked based on matchability. The system outputs the top k matches with corresponding probability of success. This match should happen relatively quickly, depending on the matching criteria that the matchmaker has defined and the amount of users in the matchmakers database. The matchmaker will provide the client with their highest matches' contact information and a summary of who they are. From here, it is on the client to contact their match and set up a time to talk or go on a date. During this dating period, the matchmaker allows the users to determine their interest in one another. However, the matchmaker may be a valuable resource for date ideas or encouragement. Regardless of the outcome of the date(s) and match, the matchmaker must be informed to obtain feedback, discuss next steps and meet with the clients for evaluation in order to determine what about the match did not align with their preferences or interests. Based on that discussion, the matchmaker will remove the clients unsuccessful match from the pool of eligible candidates and will run the system for another match. However, if the match is successful such that the individuals get married, then the matchmaker removes the users profiles from the system and has another success story, as well as a wedding invitation.

E. Parameters

a. Input

Input for the system can be split up into two different categories: implicit and explicit. Explicit input includes the data from the form given to the client and the information that the matchmaker gathers about the client from their meeting, while implicit input is the information that the system obtains about the client via web scraping using the explicit data. The form given to the client inquires about personal information, personality, family background, professional ambitions and dating preferences[1] which are explored in further detail below:

Personal Information

- Name, email, social media links, self description of looks, picture.
- Age, weight, height, ethnicity, city of residence, current relationship status.
- Education, occupation, annual income.

Personality, Lifestyle, Views

- Religion, level of observance, political inclination.
- Interests, hobbies, character traits, aspirations, lifestyle.
- Smoking behavior, alcohol consumption, drug usage.

Family Background

- Family history (origins).
- Parents and siblings information (name, education, city of residence, occupation).

Professional Ambitions

- Are they interested in returning to school?
- Do they travel frequently?
- Long term professional goals.

Dating Preferences

- What are their reasons for joining the matching services?
- Are they interested in women or men?
- What are they seeking from their partners in the short term/long term?
- What character traits/qualities are a necessity? Which are deal breakers?
- Are they only interested in dating someone within their religion?
- Are they interested in dating someone who is on a different level of observance?
- Are they interested in someone with or without dietary restrictions?
- Are they open to dating a partner with a different ethnic/racial background?
- Do they have a preference in political views?

Once the matchmaker has loaded this form into the system, the system will utilize the explicit input, such as the user's social media links, to search the web for implicit input. During this phase, the system will be able to collect information that the client might not have disclosed in their form, such as personal preferences. More detailed explanation of the web scraping process can be found in Section F: Algorithms, Knowledge Discovery. Additional input includes the matchmakers initialization criteria. Each matchmaker is unique and they match individuals based on different qualities. When the system was initialized on the matchmakers personal computer, it prompted the matchmaker to indicate what they felt was of highest importance to make a successful match. The matchmaker was able to rank qualifications that they felt should have a high weight in the matching process. For example, if the matchmaker believes that individuals are best matched based on their character traits, then the system will have character traits weigh more when assessing matchability.

b. Knowledge

Prior to running the matching algorithm, the system will have knowledge about the client which includes information gathered from explicit and implicit input. Implicit information will not always be the same for all clients - some clients might have more of a social media presence than others, thus they will have more information to be gathered. The matchmaker will have the option to input their opinion about who they think their clients ideal match is, thus aiding the system by specifying the type of person. From the initialization, the system knows the matching preferences of the matchmaker and will use those to calculate the best matches by putting a heavier weight on traits that the matchmaker thinks is important in a successful match. The system also knows the user's strict requirements and preferences for a match which it can use to draw conclusions from this data. The overall system will utilize a remote cloud platform for storage purposes on user information and each matchmaker will receive their own bucket within the cloud. This bucket will contain the data regarding the matchmaker preferences, their clients' profiles and matching history. The data collected is populated in cloud storage and processed using the high processing power of cloud computing technology in order to discover useful trends for learning. In the sense, nothing will be stored on the matchmakers local computer and it will be only used as an intermediate mechanism.

c. Output

The system utilizes explicit and implicit input in order to determine a match. The system will complete a pre-screening stage to rule out any candidates that do not fit within the client's strict requirements. From the remaining candidates, the system determines ranks the candidates by matchability. This is determined by matchmaker preferences, client preferences and feedback from prior matching criteria (as explained in Section G: Learning). Once the system has assigned each candidate a match percentage, it selects those above a certain threshold and recommends the highest ranked candidate as the client's best match. If the matchmaker agrees on the match, they will inform their client of their matches name, basic profile summary and contact information, as well as suggested next steps.

d. Assumptions

Assumptions reduce the complexity of the problem by reducing use cases, creating a more homogenous problem state. Assumptions about the processes and baseline capabilities of the system allow for the description of the best possible problem state. The assumptions that define our system and problem are as follows:

- There will be a matchmaker manning the system, whose tasks include initializing the system with their matching preferences, loading the explicit input of the client into the system, running the system and handling the matches.
- The system mainly relies on clients for information about themselves (explicit input), so it is possible that there will be gaps in the data provides.
- The matchmaker can reject clients.
- The client must accept the matchmakers dating philosophy, as it plays an important part in the matching process.
- If a client successful finds a match, their entire profile will be removed from the system thus no longer available for future matches.
- All clients have strict requirements (hard constraints) and preferences (soft constraints) for potential matches.

Many of these assumptions are made so that the system is staying with the criteria of the problem statement, such as preserving the traditional process of matchmaking while advancing and enhancing its performance.

e. Constraints

Constraints are limitations imposed by the conditions of the problem at hand, such as physical restrictions, software capabilities,and effectiveness. These constraints must be considered in order to appropriately design a solution. The constraints on our system are as follows:

- Clients will only receive one match at a time.
- Upon receiving a match, clients must at least go on one date.
- If a match is unsuccessful, the client must touch base with the matchmaker before receiving another match.
- A matchmaker will eliminate the unsuccessful candidate from the client's pool of eligible candidates.

g. Evaluation Criteria

Evaluation criteria provides the system with context of its effectiveness. In terms of matchmaking, effectiveness is determined by whether a match was successful or not. Feedback is only received at certain times within the matchmaking process and this program aims to be the least intrusive as possible, thus asking the user continuously was not a viable option. The only times that the system expects feedback are when the client requests a new match (as a result of the prior match being unsuccessful) or if a client gets married (as a result of the prior match being successful). If a new match is requested, we can assume that the previous match was not successful.

Once clients have been matched, they enter the dating phase and their profiles are put into a temporary waiting area so that they cannot be matched with other people during this time. The clients are required to go on at least one date, after which they can decide if they would like to continue or not. If they decide not to pursue after one date, the matchmaker must meet with both clients to determine why. The matchmaker provides them with a feedback form which inquires about the traits, characteristics and aspects of their match that they did not enjoy. This feedback form is primarily multiple choice which allows the eays processing for the system. The profiles of both of the clients are updated with the results from the feedback form so that the system will analyze the information from the feedback form in the next matching iteration. Finally, their profiles are removed from the temporary waiting area to be added to the matching directory. If they decide to continue dating, the matchmaker occasionally calls both clients for updates where they inquire about preferences to be added to their profile. If they ever decide to stop dating after x amount of dates, the same process with the feedback form repeats. If the clients get engaged, then the matchmaker pullback on how often they call the clients but they must still be updated with news. If the clients get married, the matchmaker confirms the match into the system and permanently closes their profiles within the system.

Success in the matchmaking process is defined as marriage. As a result, anything less than marriage is considered a “soft failure” - meaning that while they did not get married, it is possible the match was fairly successful. To measure this feedback, a scale was created to assign values to each type of feedback the program expects to receive. The scale is as follows:

- Marriage +3%
- Engagement +2%

- Greater than 5 dates +1%
- 3-5 dates 0%
- 2 dates -1%
- 1 date -2%

This scale shows the positive and negative percentages associated with the various levels of feedback. All clients are required to go on at least one date with their matches, the bottom of the scale starts with one date as being a negative percentage. As the number of dates increased, so does the percentages. These percentages correlate to the learning aspect of the matchmaking system. While this will be further explored in Section G: Learning, it is important to note that the scale affects a dynamic data structure called a word vector. Word vectors will be altered depending on the percentage values from the scale which aids the system learn from its matches. For example, the actions of 1 date and 2 dates correspond to the negative percentage values of -2% and -1%, respectively. These values subtract from the word vector values. The actions of 3 to 5 dates corresponds to a neutral percentage value of 0% which means that the word vectors will not be affected at all. The actions of greater than 5 dates, engagement and marriage correspond to the positive percentage values of 1%, 2% and 3%, respectively. These values add to the word vector values.

It is important to note that these three ranges of positive, zero and negative values were determined by the system administrators and are open to being altered. It is possible that users may see two dates as enough dates to constitute a successful match. The system will not know without processing several matches. For more, see Section I: Future Work.

F. Algorithms

a.Previous Attempts

I. SAT-Based Relational Model Finder

One of the earliest algorithms that was considered for the matching portion of the system was an SAT-based relational model finder. [4] The logic behind this SAT method included a compact representation of boolean formulas inspired by boolean expression diagrams and reduced boolean circuits. This logical representation was robust, as it provided an economical translation from relational to boolean logic which produced promising results. However, this SAT-based system produced boolean values of 0 or 1. Thus, the matchmaker would just be provided with a 0 meaning that the pair are not compatible or 1 meaning that the pair are compatible. We determined that it would be more valuable for our system to provide the matchmaker and their clients with probabilities of matchability for each possible match. This way, the matchmaker would examine matches above a threshold of a 0.50 to recommend to their client.

II. Naive Bayesian Classifier

Another algorithm that we initially considered was a Naive Bayesian Classifier which would aid in the rich profiling section of the system. This classifier incrementally learns from user feedback in order to revise user provided profiles over time. [5] This classifier is effective in a wide range of domains which is why it was significantly considered to be utilized within out system. However, this classifier puts a heavy dependence on a complete user profile. In our system, a user's profile may differ in completeness as it depends on the results of the implicit web search o the user. While we decided against this classifier, we retained its ability to refine probability within user profiles within our system.

b. Overview of Final Design

Overall, this system compiles the clients explicit and implicit data from their user profile to compose a pool of eligible candidates from the matchmakers directory of clients, ranks these candidates by matchability, sends this ranking to the matchmaker who will inform their client of their highest match in order to commence the dating process. *Figure 1* shows a high level overview of the matchmaking process utilizing the system. Starting from the left side of the diagram, the matchmaker downloads the system onto their personal computer and they initialize the system with their matching preferences which reflectings their personal matchmaking philosophy. Next, their client fills out an extensive form which includes information regarding themselves (as detailed in Section D: Parameters, Input). The matchmaker loads this form into the system and the system utilizes this explicit input to gather implicit input about the client using a web scraping technique. This combination of input produces a user profile of the client which is stored into the matchmakers directory of clients. Using the strict requirements defined by the client, the system queries the directory of total users to produce a pool of potential matches. Using the dating preferences defined by the client and the matchmakers preferences defined by the matchmaker during the initialization of the system, the system ranks the users within the pool of potential matches. This ranking is returned to the matchmaker by the system, who is in charge of approving the highest ranked match and providing the client with their matches contact information, basic profile summary and suggested next steps in the process. During this dating phase, the profiles of both the clients are placed into temporary waiting area so that they cannot be considered by the system for other matches. If the dating phase is successful, then the matchmaker removes the profiles of both the clients from the directory. If the dating phase in unsuccessful, the matchmaker must meet with the client in order to collect feedback for the system to enhance the next match their receive. This failed match will not be considered for future matching by the system and the process will repeat for the client starting at ranking the pool of eligible candidates. Despite the outcome of the match, the system utilizes the characteristics and traits of the failed match to assemble and adapt the desired characteristics and traits of the client in order to enhance the ranking of future matches for the client.

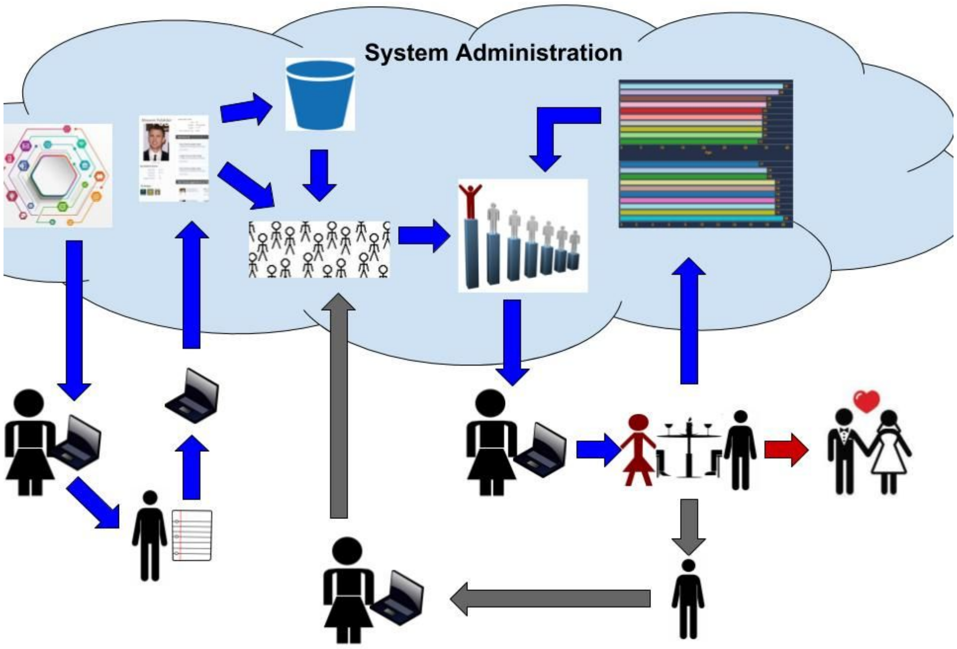


Figure 1: High Level Overview of Overall System

c. Knowledge Discovery

In order for the system to make informed decisions, it must be equipped with a defined set of initial knowledge. In Section E: Parameters - Knowledge, established that the knowledge of the system would contain both explicit and implicit user input, as well as any other information the matchmaker regarding the user that the matchmaker inputs into the system. The explicit input is collected via an extensive survey, comprised mostly of multiple choice questions. The implicit input is collected via a web scraping technique known as, human copy paste [2]. In this section, we further explore the semantics regarding the different types of knowledge discovery within the system.

I. Data Collection

As discussed in Section E: Parameters - Input, the explicit input is an extensive form provided by the system for the client which inquires about personal information, personality, family background, professional ambitions and dating preferences. The majority of these questions are multiple-choice, thus are categorical questions with a predefined set of answers. Categorical questions make it easy for the system to collect answers. For example, when inquiring about smoking behavior, the multiple-choice options include ‘Never’, ‘Rarely’, ‘Sometimes’, ‘Often’ or ‘Every Day.’ Once the client has filled out this extensive form, they meet with the matchmaker to review it. This is an opportunity for the matchmaker to pull more information that the client did not note or identify additional qualities about the client that they think are important to note. This explicit input is added to the clients user profile within the systems database, which is stored inside a private cloud system. Utilizing the user's explicit input, the system will be able to gather implicit input about the user.

Implicitly gathering input on a user is a task that would aid the system improve its matching abilities. The data about a user will be gathered explicit through user and the matchmakers input until it is determined that there is enough explicit data to commence an implicit search. Once the web search for the user is complete, the data gathered explicitly will be used to verify the implicitly collected data to check its validity. In this way, the system can still reliably explicitly gather information, while simultaneously building a parallel system to collect data implicitly. This is an important matter because it reduces the instrusivity involved in the explicit data collection, where the client must fill out an extensive form on themselves and fill out feedback forms for the system if matches are not successful. The systems method of gathering data implicitly relies on three commonplace techniques when web scraping:

1. Human Copy-Paste [2]
2. Regular Expressions [6]
3. Semantic Annotation [7]

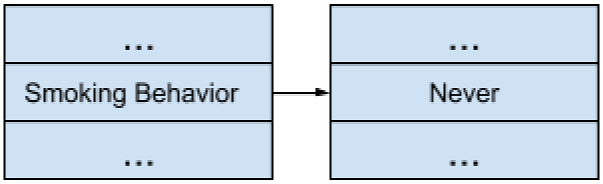
Given a webpage that the system determines should be searched for information, the first step would be to use human copy-paste to pull all of the text from the page. [2] Once this text is extracted from the site, it is saved and ready to be analyzed. The second stage involves using regular expressions to find keywords that the system hopes will identify a sentence as containing important information. [6] The sentences that are identified as containing these keywords are then set aside for the final stage. The final stage uses semantic annotation to identify the important information relating to keywords. The process of semantic annotation is the act of labeling words in a sentence and relating them to each other in order to make sense of the sentence and its context. [7] This final stage will help the system derive information from each sentence in hopes of filling up a portfolio on the client.

d. Data Representation and Organization

Once the data regarding a user has been collected, it must be properly stored somewhere. In Section F: Parameters - Knowledge, it was established that the system will utilize a remote cloud platform for storage purposes. Within this cloud system, each matchmaker will receive their own bucket to store data in. This data will mainly comprise of user information, matchmaker preferences and past matches.

I. User Profile

Within the bucket, information regarding users will be stored in the form of a profile. A profile contains essential information about an individual user[3]. Users will differ in their strict requirements and dating preferences for matches. For example, one user may only want to date someone of a particular religion, while another user may be open. The user profile is vital in providing the users with personalized matches. We have previously noted that the content of the user's profile will be primarily explicitly collected and secondarily implicitly collected. Within a user profile, the categorical information from the form will be stored in a data structure called a hash table. These accessible and fast structures allow the system to store a key with a corresponding value. For example, if a user selected ‘Never’ for the question which inquired about smoking behavior, then the hash table entry for this question would resemble the diagram below:



The profile itself will be an object created at the time of the user's initialization into the system, which is when the matchmaker loads their explicit information into the system. The profile will be adjusted as the system learns more about the user during the matchmaking process. The user will have defined strict requirements, such as requiring someone of a certain faith and within a certain age range, but a user's dating preferences can be expanded as they date. Through the feedback forms that the user fills out about failed matches, the system can adjust the user's preferences. These preferences will be stored as topic hierarchies [3] where each node in the hierarchy represents a preference for the user. The system will look to these preferences in order to rank potential matches. If a user has 'Blonde' as a node high on their topic hierarchy, the system will give greater weight to matches who are blondes. These weights will come together to form the matchability percentages for the potential matches.

II. Database of Users

A user's profile will be stored within the matchmakers directory. As previously stated, each matchmaker will be allotted a single bucket within the systems cloud system. Within this bucket, the matchmakers directory of clients will be stored. This database of clients will store the profiles of all the clients and will be easy to query. The system will utilize the users defined strict requirements to query all users that fit those requirements. For example, if a client wants a 'Man', 'Catholic', 'between 25-30 years old', then the system will query the database of all available users for meet these requirements. This query forms a pool of eligible candidates for the user. The system will rank these candidates using the users dating preferences and will return the ranking of candidates above a certain threshold to the matchmaker.

Each matchmaker will have their own bucket within the cloud because they have their own clientele within their community. This cloud system will be similar to Amazon S3 where system administrators handle cloud infrastructure, flexibility and durability. Each matchmaker using the system on their personal computing devices will handle the matchmaking process on their computers but the calculations and storage will occur within the cloud.

e. Preprocessing

This preprocessing stage occurs after a user's profile has been created and added to the database of clients. It mainly pertains to utilizing the users strict requirements (hard constraints) defined in their explicit input to query the database of clients for users that fit those strict requirements. This forms a pool of eligible candidates who fit the hard constraints. From these, the users dating preferences (soft constraints) are used to rank the candidates within the pool. Both hard and soft constraints differ by user.

I. Hard Constraints

The hard constraints are the strict requirements identified by the client. These includes requirements that a possible match must fit. For example, a client may identify that they must only be matched with someone of a certain religion, within a certain age range or of a specific race. Any characteristic or trait listed under hard constraints must be satisfied by all potential matched. As previously stated, the system will query the database of users with these requirements in order to obtain a pool of eligible candidates. These eligible candidates meet every requirement listed by the client. To see an example of how these hard constraints are implemented, see Section F: Algorithms, Main System, Matching Algorithm

II. Soft Constraints

The soft constraints are the dating preferences identified by the client. These include preferences that are not necessary for a potential match, but are preferred by the client. For example, a client may identify that they prefer someone of a certain hair color or occupation as a match. These constraints are not necessarily guaranteed to appear within a match but they affect the ranking that a match will obtain. If a potential candidate meets the clients hard constraints, and if they meet most of the clients soft constraints, then they will be ranked a higher match probability than a candidate that meets the clients hard constraints but does not meet most of the clients soft constraints. To see an example of how these soft constraints are implemented, see Section F: Algorithms, Main System, Matching Algorithm.

f. Main System

I. Matching Algorithm

The matching algorithm is the center of the matchmaking program and it is utilized to assign two people together given their strict requirements, dating preferences, and user profiles. The matching algorithm that this system uses is a derivative of a combination of algorithms, namely Description Logics, Logic Programming, and Fuzzy Theory[8]. Each of these algorithms serve its purpose and work together to create what this system refers to as its Knowledge Base. [8] For this example, refer to the two tables below. The first is the database of men currently stored in the matchmakers system and the second is the database of women currently stored in the matchmakers system:

ID	Name	Hair	Religion	Age	Income	Ethnicity	Weight
24	Jeff Love	Blonde	Catholic	28	92,500	White	210
321	Levi Berg	Brown	Jewish	32	153,000	White	185

ID	Name	Hair	Religion	Age	Income	Ethnicity	Weight
33	Jen Li	Blonde	None	23	33,400	Asian	105
121	Cat Smith	Brown	Catholic	27	103,000	White	115

In order to explain the matching algorithm in the best manner possible, an example will be explained simultaneously. This example will use a Knowledge Base K = {F, O, P} where F refers to a set of facts that the system has on that person. A set of facts is simply the information that can be gathered from that individual's table entry. For this example, the system will be attempting to match Cat Smith. F for Cat Smith would be:

Women(121, Cat Smith, Brown, Catholic, 27, 103,000, White, 115)

O refers to the Description Logic, or DL, component of the Knowledge Base. This is the only part of K that remains optional. The purpose of O is to further define values that each table might hold. A common example could be:

Catholic is a subset of Christianity.

The last part of K, the Knowledge Base, is P, the rule component. The rule component is where “rules” or preferences for that user will be stored and used in the matching process. Although P is considered the singular rule component for a user, it has two parts to it, the hard constraint rules and the soft constraint rules. Refer to Section E: Preprocessing for more information on hard and soft constraints. Essentially, they are the preferences of the user. They are differentiated by “must-have” as opposed to “preferred”. Hard constraints being the “must-have” and soft constraints being the “preferred”. Here is an example of a possible list of constraints for Cat Smith:

- ***Hard Constraints:***
    - ***Must be Jewish or Catholic.***
- ***Soft Constraints:***
    - ***(1) I would like him to make over 100k if he is over 30.***
    - ***(2) I would prefer him under 200 lbs.***
    - ***(3) His hair should be Blonde.***

These constraints can then be translated into a more formal definition for P, the rule set, for the Knowledge Base, K:

Wa(x) = Jewish(x)

Wa(x) = Catholic(x)

W(x) = Wa(x)

W1(x, a) = Income(x), Age(a)

W2(x) = Weight(x)

W3(x) = Hair(x)

The first three lines represent the hard constraints with the following three representing the soft constraints. Each function W…(x) will return a value of 0 or 1 indicating whether that constraint is fulfilled or not. For instance the first rule, Wa(x), will check to see if the current man the system is trying to match Cat Smith with is Jewish by retrieving that information from the religion column of the table. Since this is a simple yes or no answer one can easily see how Wa(x) will return a 0 or 1. For a more complex function like W2(x) the Weight(x) function will grab the weight from the table but then in the W2(x) function it will compare it to the desired value, less than 200 lbs, and then return a 0 or 1 from there. Lastly, this step combines the hard constraints into one rule, W(x), a similar process is applied to the soft constraints below:

$$Woman(x, a, m) = W1(x, a), W2(x), W3(x), m = .5 * s1 + .4 * s2 + .1 * s3$$

At this step the soft constraints are combined into one complete function. As one might have noticed a new equation is added to the later part of the new combined formula this equation is reliant on weighing each soft constraint. This would be a question that the system would have to ask the user but what it would ask would be to weigh each soft constraint out of 100% given that the total combined percentage must total 100%. In this case Cat Smith decided to weigh soft constraint 1 with .5, soft constraint 2 with .4, and soft constraint 3 with .1, totally 1 or 100%. The weight of each soft constraint will be multiplied by the probability outputted from that soft constraint equation giving a value m of the combined probabilities. This same process will be completed for the male resulting in two functions, one for hard constraints and one for soft constraints, as well:

Hard Constraints: M(x)

Soft Constraints: Man(x, a, m)

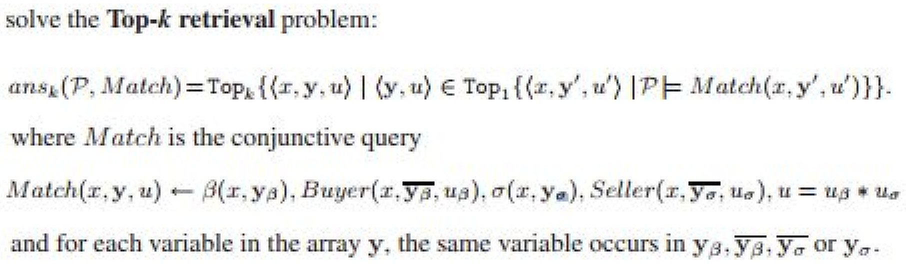
The male and female function will then be combined into one all inclusive function for their match:

$$Match(x, a, m) = W(x), Woman(x, a, m1), M(x), Man(x, a, m2), m = m1 * m2$$

The final step of the matching process is then to solve the Top-k retrieval problem with this function:

$$ans_2(K_{match}, Match)$$

Where K_{match} is the Knowledge Base from the start of this problem and the Top-k retrieval problem is defined as:



solve the **Top-*k* retrieval** problem:

$$ans_k(\mathcal{P}, Match) = \mathrm{Top}_k\{\langle x, \mathbf{y}, u\rangle \mid \langle \mathbf{y}, u\rangle \in \mathrm{Top}_1\{\langle x, \mathbf{y}', u'\rangle \mid \mathcal{P} \models Match(x, \mathbf{y}', u')\}\}.$$

where $Match$ is the conjunctive query

$$Match(x, \mathbf{y}, u) \leftarrow \beta(x, \mathbf{y}_\beta), Buyer(x, \overline{\mathbf{y}_\beta}, u_\beta), \sigma(x, \mathbf{y}_\sigma), Seller(x, \overline{\mathbf{y}_\sigma}, u_\sigma), u = u_\beta * u_\sigma$$

and for each variable in the array $\mathbf{y}$, the same variable occurs in $\mathbf{y}_\beta, \overline{\mathbf{y}_\beta}, \overline{\mathbf{y}_\sigma}$ or $\mathbf{y}_\sigma$.

This will result in probabilities that correlate to each individual that we try to match Cat Smith with creating a table like the one below where the ID of the individual is accompanied by the probability that this will be a successful match.

ID	Probability
24	0.3010
321	0.1885

G. Learning

a. Keyword vectors

The most exciting part of this matchmaking system is the the ways in which matches are improved through the learning process. The main method of learning employed in this system is using word vectors[9] to build rich profiles for each characteristic of a person. There will be two sets of word vectors created, one set for men interested in women and another set for women interested in men. It is very realistic to assume that women do not share the same interests in men as men do in women. For this reason, the system will have two sets of word vectors. The word vectors will each represent a characteristic of that person and in that word vector will be each characteristic of the opposite sex. The administrators of the program will determine ahead of time an all inclusive list of characteristics that will be used to describe a man or woman. Some of these characteristics could be, but not limited to:

- Rich
- Blonde
- Athletic
- Brunette

These characteristics and more can all be attached to a user to identify them in the system. As previously mentioned, each word will have a vector associated with it containing the full list of characteristics for the opposite sex. The word vector for Rich will look like the following:

RICH

Rich	Blonde	Athletic	Brunette	...
.5	.5	.5	.5	...

The "Rich" word vector holds every characteristics in the list determined by the administrators. Accompanying those words is the probability that a "Rich", in this case, woman will like a Rich, Blonde, Athletic, etc. man. These values are originally set to .5 or 50% as it is the start case to be refined over time. Now to put the full list of word vectors in perspective, this is what the full list of word vectors for a woman will look like:

RICH

Rich	Blonde	Athletic	Brunette	...
.5	.5	.5	.5	...

BLONDE

Rich	Blonde	Athletic	Brunette	...
.5	.5	.5	.5	...

...

...	...	...	...	...
...	...	...	...	...

Each characteristic for a woman will have its own word vector with every characteristic of a man represented in it along with the associated probability that that kind of woman will like that kind of man. The men will have an identical but separate list of word vectors representing the likelihood that that type of man will like that type of woman. The reason why the system was designed this way was so that the system could, over time, figure out which type of man likes which type of woman and vice versa so that depending on the characteristics of a new client the system could assign them a match with a high probability of ending in a successful match. The next two sections detail how these probabilities, initially .5, are modified over time and how they factor into the matching algorithm.

b. Updating keyword vectors

Keyword vectors are updated based on the feedback the system receives. Feedback is detailed in Section E:g. Refer to that section to see the percentage change for each word vector based on the feedback. Now that one is aware of what feedback results in what values the next step is to understand how each value is being modified. Take the example of Cat Smith from the matching algorithm section. At the conclusion of that example she ended up matching with Jeff Love. For the sake of this example, assume that they end up getting married and the system is provided that feedback. The system now has to parse through the word vectors and update their values based on the feedback. Cat Smith had Brown hair and was Catholic while Jeff Love had Blonde hair and was Catholic. The algorithm will search through each word vector associated with Cat and then Jeff starting with Brown (female), then Catholic (female), then Blonde (male), etc. Starting with Brown for female hair the system takes every characteristic from the male, Jeff Love in this example, and modifies their probability by the feedback value. The system uses two formulas for this:

For an increase: CurProb + (ModProbe * (1 - CurProb))
For a decrease: CurProb * (1 - ModProb)

These formulas will successfully clamp the probabilities so that they cannot drop below 0% and cannot rise above 100%. So for the list of female vectors the first to modify was the Brown (hair) word vector. Before the changes, since this was the first run of the program, the Brown word vector looked like so:

BROWN

Rich	Blonde	Athletic	Brunette	...
.5	.5	.5	.5	...

After the match is deemed successful and the feedback is recorded by the system the word vector will look like so:

BROWN

Rich	Blonde	Athletic	Brunette	...
.5	.515	.5	.5	...

This was calculated from this formula: ***CurProb + (ModProb * (1 - CurProb)) = (.5)+ ((.03) * (1 - (.5)))***

Interpreting this from a glance, this means that a Brown haired female is more likely to successfully match with a Blonde haired male. This system was designed so that matches could be given based on results from prior attempts after determining what was successful or not. The hope is that over time the probabilities will eventually stabilize and the system will be able to identify trends and common matches based on characteristics which will in turn help the system to make more accurate matches. This process of modifying the probabilities will continue for each word vector that applies to that person and then it will repeat the same process but with the other match.

c. Utilizing these to enhance matches

The last part of the learning section is how this process of using rich profiling with word vectors will enhance matches in the future. The application of this is quite simple. Each probability for a given characteristic will be introduced as a soft constraint into the set of rules, P, for the matching algorithm. Let's stay with Cat Smith for this example to maintain consistency. When the system starts trying to build P, it will cycle through every word vector for females that apply to Cat Smith, such as Brown (hair), Catholic, etc. It will look at the characteristics assigned to the male that is attempting to be matched with, for Jeff Love, Blonde (hair), Catholic, etc. It will then extract the probability from each word vector that applies to Cat associated with Blonde (hair) and then average them together. Then it will do this but for Catholic. Once this process is complete the system will have an average probability spanning all word vectors tailored to that person and their attempted match. This will seamlessly incorporate into the soft constraints section where it will refine matches based on what has previously been learned.

H. Algorithm Evaluation

The most exciting prospect about this matchmaking system is the rich profiling to create a result-based matching system. As stated multiple times before, the matchmaking process is a time-honored tradition which has been practiced for generations without utilizing technological advances. This system challenges the current traditional practices in order to maximize successful matches and productivity. The algorithms chosen work seamlessly together and enable this technique to come to fruition. The matching algorithm that was chosen employs a set of rules, which is where hard and soft constraints can be defined by a client based on their requirements and preferences. In addition, the structure of the soft constraints makes it the ideal place to add in probabilities from the word vectors to factor into the matching process. Finally, the word vectors are like mentioned previously where this system really propels the matchmaking process forward. A results-based approach to matching will most likely show a trend over time and as a result successful matches will be found quicker than normal, which at the end of the day will improve the matchmaking process as a whole and push the field forward.

I. Future Work

Below are some ideas that we have to further enhance the system in the future:

- Implicit data collection using image detection: The system could scan a client's social media page for images of them. If the system notices that the client has many pictures with brunettes, then the system would assume that the client prefers brunettes thus would place greater emphasis on matching the client with a brunette. For this, we would need access to their social media accounts and a sophisticated image recognition system.
- Connect matchmakers of similar traditions around the world: As more matchmakers turn to our system to enhance their process, the system will gain recognition and its network will expand. The system could adjoin directories of matchmakers from similar traditions or faiths in order to expand a clients options. By doing so, the system would be developing an extensive network of matches and would be putting together international matches.
- Remove reliance on matchmaker: As the system puts together more matches, it will learn the preferences of a wide range of people. This way, the system could communicate directly with the clients themselves, rather than using the matchmaker as a middle man.

J. Conclusion/Ethics

Artificial Intelligence has been regarded as a double-edged sword considering it has the power to greatly aid yet severely hurt people and society. Before implementing and deploying a system, it is imperative to consider the advantages and disadvantages that the system may bring about. In matchmaking systems, there is a focus on developing an application that is a hybrid of current technology and old-fashioned matchmaking. Creating such a system can aid matchmakers keep up with the demand while utilizing artificial intelligent techniques that can enhance their abilities will benefit all parties involved.

The utilization of a system that improves successful matches made by a matchmaker will be chiefly beneficial to those seeking a matchmaking service. These clients will be able to receive personalized dating advice from a real person who has been in this business many years. Compared to an online dating website, the matchmaking services are more productive for clients as they do not have to spend countless hours sifting through profiles on dating websites or trying to connect with the people they find. The matchmaker sifts through profiles and makes the connection for them. A matchmaker can be considered a wing woman (or man) so that the clients are not in the dating scene alone, thus making is a valuable learning experience for them. With the clients preferences in mind, the matchmakers will pre screen the candidates which will result in higher quality matches. Finally, the client will be able to meet their matches in person as matchmakers have clientele located in a specific area. On the other hand, there are many aspects of such a system that concerns many people. IT will speed up the matchmaking process, thus the matchmaker might be making less money as they are able to set up matches quicker. The system cannot guarantee that a client will find a match, especially if the matchmaker has a limited database of available candidates. While this possibility may occur far ahead in the future, it is possible that advanced systems like these replace the need for experienced matchmaking professionals. Doing so would change the nature of a community.

Overall, perhaps implementing this system is the natural progression of the matchmaking field and the dating industry. Using this system, matchmakers will be able to make data and statistics driven decisions. They will be provided with the proper evidence to form a match and will be able to make successful matches within their communities. The system will expedite the matchmaking process so that more clients can be served, resulting in lower costs with better outcomes. Finally, as the data grows and the learning algorithms improve, more clients from a wide range of backgrounds and profiles will be able to benefit from the collective knowledge gained from the data. In conclusion, we believe that our system efficiently provides a helpful mechanism to aid matchmakers and those seeking love.

K. References

[1] Hitsch, Günter J., et al. *“What Makes You Click?-Mate Preferences in Online Dating.”* SpringerLink, Springer US, Quantitative Marketing and Economics, 20 May 2010.

[2] Vargiu, Eloisa; Urru, Mirko, et al. *“Exploiting web scraping in a collaborative filtering-based approach to web advertising.*” Sciedu, Artificial Intelligence Research, 5 December 2012.

[3] Amandi, Analia; Schiaffino, Silvia, et al. *“Intelligent User Profiling.”* SpringerLink, Springer US, International Federation for Information Processing, 2010.

[4] Torlak E., Jackson D. (2007) *“Kodkod: A Relational Model Finder.”* In: Grumberg O., Huth M. (eds) Tools and Algorithms for the Construction and Analysis of Systems. TACAS 2013. Lecture Notes in Computer Science, vol 4424. Springer, Berlin, Heidelberg

[5] Pazzani, Michael; Billsus, Daniel. *“Learning and Revising User Profiles: The Identification of Interesting Web Sites.”* Kluwer Academic Publishers Machine Learning (2009) 27: 313.

[6] Munzert, Simon; Rubba, Christian; Meibner, Peter; Nyhis, Dominic. *“Automated Data Collection with R: A Practical Guide to Web Scraping and Data Mining”*. John Wiley & Sons Ltd., 2015

[7] Malik, Sanjay Kumar; Rizvi, Sam. *“Information Extraction Using Web Usage Mining, Web Scraping and Semantic Annotation.”* IEEE, CICN: International Conference On Computational Intelligence and Communication Networks, 2011.

[8] Ragone A., Straccia U., Di Noia T., Di Sciascio E., Donini F.M. (2007) *“Vague Knowledge Bases for Matchmaking in P2P E-Marketplaces.”* In: Franconi E., Kifer M., May W. (eds) The Semantic Web: Research and Applications. ESWC 2007. Lecture Notes in Computer Science, vol 4519. Springer, Berlin, Heidelberg

[9] Pennington, Jeffrey, Richard Socher, and Christopher Manning. "Glove: Global vectors for word representation." Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP). 2014.