Therapy Recommendation System

Md Azgar Hossain Shuvo
University of Trento
1st Year, Computer Science
mdazgarhossain.shuvo@studenti.unitn.it

1 INTRODUCTION AND MOTIVATION

According to various symptoms at different levels of a patient, it is highly crucial to suggest a compatible therapy for a disease. The accurate classification of different therapy is essential in helping doctors carry out compatible treatment schemes for the disease. Usually doctors are prescribing therapy to the patient using their own knowledge and previous experiences. To prescribe a therapy for a certain patient a Doctor always needs to take a consideration not only the patient's previous condition but also the previous therapy. Further, a doctor always consider the patients previous medical history, symptoms, and many more factors. Since, some therapy could be ineffective for a particular patient, some are time consuming as well as costly, more importantly some therapy is harmful for a specific patient, that's why it's not always easy to provide best therapy for a doctor. In 2018, a study showed more than 250,000 people in the U.S. die every year from medical errors. Other reports claim the numbers to be as high as 440,000 [6]. Considering these issues we used some conventional data-mining techniques to recommend best therapies that would be effective for patients. In order to attain this, we merged patient conditions with patient filtering with the therapy success rate and formulate a Patient-Trial interaction matrix (PTI).

2 KEYWORDS

Recommendation System, Item-item collaborative filtering, Utility Matrix, Normalization, Nearest Neighbor, Clustering, Evaluation, Regression, Root Mean Square Error, Data Mining,

3 RELATED WORK

In this section we briefly describe some of the research literature based on recommender system in healthcare as well as collaborative filtering. The number of choices can be overwhelming in a world, where users find and evaluate items of interest by the help of recommender systems. By associating the element of recommended items or the opinions of other individuals, they connect users with items. For example, a consumer of just about any major online retailer who expresses an interest in an item – either through viewing a product description or by placing the item in his "shopping cart" – will likely receive recommendations for additional products [5] . By the demographics of the customer or the top overall sellers on a site, these products can be recommended.

Using big data, people accept the way of treatment day by day. For instance, Steve Jobs has chosen the suitable treatment plans according to data mining of his gene. In internet industry, the big data of healthcare has gradually become one of the sophisticated one. The goals of healthcare recommendation systems are built on promoting personal healthcare in a few areas, including life, psychology, and career. It is challenging to recommend relevant services that are suited for the users and satisfy the users' objective needs when recommendation algorithms lack a thorough grasp of the users' health situation. Additionally, the conventional approaches that develop item models cannot reflect item feature due to the more serious issue of sparse healthcare data. The recommendation method will always have a strong bias because the understanding of the users' preferences is insufficient. Suggesting a recommendation system built on a human-centric approach to address this issue. In order to offer individualized recommendations, the system determines the user's health attributes based on their physiology, beliefs, character, experience, knowledge, and surroundings. [7].

Both the expense of medicine and the standard of healthcare must be reduced in society. This manuscript suggests the creation of substitutes for the conventional healthcare methods of disease prevention and ongoing health monitoring. Having affordable access to the greatest care and the ability to prevent sickness can give each person power and enable them to a longer, healthier life, and improved performance on the individual level. These emphasize using data mining to identify user preferences and provide individualized healthcare services like disease monitoring, treatment, and rehabilitation [2].

Finding user groups that seem to share similar preferences is how clustering techniques operate. Predictions for an individual can be established by averaging the feedback from the other users in a cluster after it has been created. Each user is portrayed by some clustering approaches as having varying degrees of membership in various clusters. Next, a weighted average of the predictions from each cluster is calculated. When compared to other methods, clustering techniques typically result in less personalized recommendations, and in certain circumstances, the accuracy of the clusters is even poorer than that of closest neighbor algorithms. However, performance can be quite good once clustering is finished [3].

It is now time to develop the tools that will enable us to sort through all of the information at our disposal and identify the data that will be most beneficial to us. Collaborative filtering is one of the most promising of these technologies. Building a database of user preferences for things is how collaborative filtering operates. There has some fundamental challenges for recommender systems which is implemented with collaborative filtering such as scalability, quality, runtime, and performance. These challenges make some conflictions. Using the item based collaborative filtering algorithm these problem can be solved. The conventional collaborative filtering algorithm search for neighbor among a large number of populations or users. But item based collaborative filtering avoid this technique by building up the relationship between items first rather than users [1]. We decided to use Item based collaborative filtering algorithm in our recommendation system and observe how therapies are work on particular patients analyzing their conditions followed by their previous medical history.

© 2021 Copyright held by the owner/author(s). Published in Proceedings of the 24th International Conference on Extending Database Technology (EDBT), March 23-26, 2021, ISBN 978-3-89318-084-4 on OpenProceedings.org.

4 PROBLEM STATEMENT

The patient dataset is one of the crucial part of our problem. We worked with a patient dataset which is formatted with JSON. A patient is an entity who has a unique identifier as well as a collection of attributes in ["name":"Value"] pair. These patient is also have some other attributes such as (Conditions, Therapies), both having several attributes with key:value pairs. An illness is respect to a condition and doctor have designed therapies to treat them. These conditions can be short term or long term. When a patient suffering from illness, doctors suggest therapy checked by the condition. A trial has been applied to a specific patient for a particular condition, that is called therapy. The trial which is a tuple <t, p, date, params, success>. Here, t is a therapy, p is a patient, date is applied as a time, and params is a set of attribute in <name:value> pairs that describe the therapy. A patient has a condition. Under every condition a sequence of therapies are suggested as trials. These trials are performs in every patients differently based on specific condition. In trials, there is a parameter named successful which indicates the percentage of the efficiency of a therapy for a specific condition. May a patient don't have a trial if a condition is found but not treated yet. Therefore, we have implemented a recommender system that will suggest a efficient therapy for a patient's trial best matched with the specific condition which is uncured. Given the following **inputs**:

- A list of patients **P**, with a set of conditions, trials and therapies
- A specific patient and his/her medical history $\mathbf{P}_h$
- A Particular condition **C**

The **output**:

- A list of recommended therapies $\mathbf{T}_h$

In addition, we have to implement a baseline method to compare our method as well as determine whether the proposed approach produces better recommendation.

5 SOLUTION

We should consider that, if a therapy is successful for a patient respect to a particular condition that therapy may not be successful for other patient. Based on the success rate of one patient, we can not suggest the same therapy to other patient. In order to have a outstanding prediction for every patient. our implementation should follow below aspects:

- Find the matched therapies which have the similar/closed success rate
- How successful a therapy is for a condition
- How well a patient responds to a therapy

To implement these above steps we followed the approach of item-based collaborative filtering to find the correlation between a patient and a therapy because it shows better outcomes as compared to the user-based collaborative filtering. We need to select a set of condition and trial combinations for a patient based on the trials that have a higher success rate. Our proposed solution will address the following steps:

- Merge Trial with Condition
- Patient-Therapy interaction matrix
- Selecting global best therapy
- Condition-therapy interaction matrix
- Nearest neighbour
- Correlation between top therapy and patient specific therapies
- Cold start problem
- Evaluation

5.1 Merge Trial with Condition

Mainly, we have three entities in our dataset such as (Patient, Condition, therapy). Under patient we have two entities such as (conditions and trials) where condition and trials connected with patient condition id. That's why We merged the trials data with condition data matched with identifier. As a result we get all of the trials data in respect to the conditions in a single data frame.

5.2 Patient-Therapy interaction matrix

Based on the therapy success rate we creates a interaction matrix between patient and therapy. This matrix gives us a relation between individual patients and individual therapies. It provide an overview on the best therapies ranking and will be useful to calculate correlation.

5.3 Selecting global best therapy

To compare other therapies we have to make a specific therapy as a global which success rate is highest. In addition, a new patient has no previous record. So we are finding the global best successful therapy for the new patient checked by specific condition and suggest that.

5.4 Condition-therapy interaction matrix

We creates an interaction matrix between condition and therapy based on the success rate is considered. This matrix can provide an insight on the best therapies ranking for a specific condition

5.5 Nearest neighbour

We will apply the Euclidean Distance formula from the condition-therapy interaction matrix for nearest neighbors to calculate the therapy ranking based on success rate for each condition

$$d(x,y) = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2}$$

5.6 Correlation between top therapy and patient specific therapies

A global top therapy is selected filtering with several patient conditions. A correlation is calculated with other specific therapies compared with this top therapy. The Pearson correlation coefficient formula is given below:

$$r = \frac{\sum(x_i - \overline{x})(y_i - \overline{y})}{\sqrt{\sum(x_i - \overline{x})^2 \sum(y_i - \overline{y})^2}}$$

Notations:
r = correlation coefficient
x_i = values of the x-variable in a sample
$\overline{x}$ = mean of the values of the x-variable
y_i = values of the y-variable in a sample
$\overline{y}$ = mean of the values of the y-variable

5.7 Cold start problem

We are considering the cases where we can find the correlations among conditions, patients, and therapies. These selective cases have previous medical history or conditions. Given the below list, we need to address cold start problem.

(1) A new condition that was never treated

(2) A patient who has conditions but no trials
(3) A patient who has no condition
(4) A new patient with no records

To formulate these cases we need to find the most successful therapy from the data of condition-therapy interaction matrix. Also finding the similarities with the data from patient-therapy interaction matrix and predict best therapy.

5.8 Evaluation

We are using Root Mean Square Error (RMSE) to evaluate the regression problem of our algorithm where the output (predicted value) is compared with actual value for a given data points. Given the below formula we are using to calculate RMSE:

$$RMSE = \sqrt{\frac{1}{|R|} \sum_{(i,x) \in R} (\hat{r}_{xi} - r_{xi})^2}$$

Here,
R = the number of data points
$\hat{r}_{xi}$ = predicted value
r_{xi} = actual value

For evaluation, initially we take 5 entity randomly form the patients dataset which condition is cured. Then select the therapy success rate from the merged dataframe corresponding to the selected random entity. Then we use our recommendation algorithm to take the predictive value for selected data. After that, we use the Root Mean Square formula to find the deviation between predicted ratings and actual ratings value. We repeat the RMSE formula two times for further records.

6 IMPLEMENTATION

To implement our solutions we used python as a programming language to build the recommendation system. The python libraries we are using are json, pandas, numpy, traceback, math. The blueprint of our implementation is stated below based on above solution:

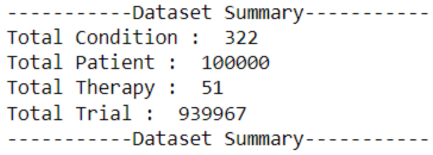
- Plugin and summarize dataset
- Fetching condition id
- Creating Dataframe
- Generating top therapy
- Calculating mean success rate for trials
- Calculating number of successful trials
- Generating patient-therapy interaction matrix
- Finding correlation with other therapies
- Joining number of successful trials
- The final recommendation

6.1 Plugin and summarize dataset

We plugin the provided patient dataset with development environment through file path. Then we summarize the dataset to have an overview of Conditions, Patients, Trials, Therapies After taking a overview of the whole dataset we saw it consists of 322 conditions, 100000 patients, 51 therapies, and 939967 trials (Figure 1)

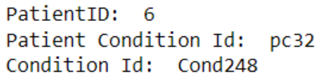
6.2 Fetching condition id

We fetch the condition Id by a method which take two parameters. One is the patient id and another one is the patient condition id. With these two parameter we fetch the original condition id from condition dataframe (Figure 2).



```
-----------Dataset Summary-----------
Total Condition :  322
Total Patient :  100000
Total Therapy :  51
Total Trial :  939967
-----------Dataset Summary-----------
```

Figure 1: Dataset Summary

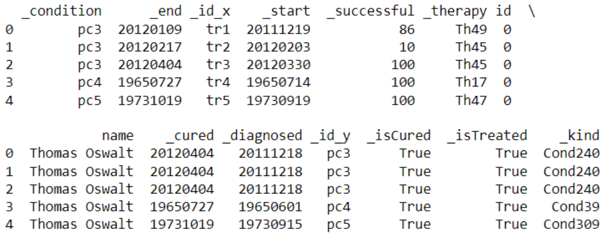


```
PatientID:  6
Patient Condition Id:  pc32
Condition Id:  Cond248
```

Figure 2: Fetching condition id

6.3 Creating Dataframe

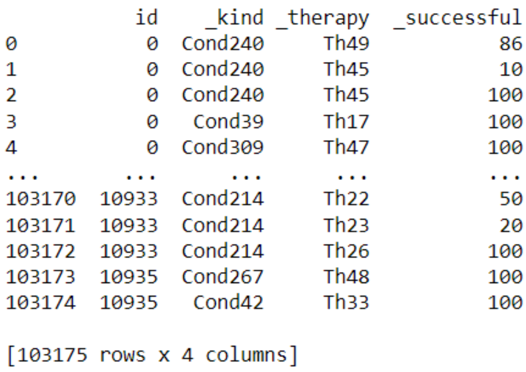
The provided dataset is JSON format consist with condition, patient, therapies. Thinking about readability and scalability, we convert it into a dataframe (Figure 3). After that we filter the dataframe by setting some columns fixed (Figure 4).



```
  _condition      _end _id_x    _start  _successful _therapy id  \
0        pc3  20120109   tr1  20111219           86     Th49  0
1        pc3  20120217   tr2  20120203           10     Th45  0
2        pc3  20120404   tr3  20120330          100     Th45  0
3        pc4  19650727   tr4  19650714          100     Th17  0
4        pc5  19731019   tr5  19730919          100     Th47  0

           name    _cured _diagnosed _id_y  _isCured  _isTreated    _kind
0  Thomas Oswalt  20120404   20111218   pc3      True        True  Cond240
1  Thomas Oswalt  20120404   20111218   pc3      True        True  Cond240
2  Thomas Oswalt  20120404   20111218   pc3      True        True  Cond240
3  Thomas Oswalt  19650727   19650601   pc4      True        True   Cond39
4  Thomas Oswalt  19731019   19730915   pc5      True        True  Cond309
```

Figure 3: Creating Dataframe



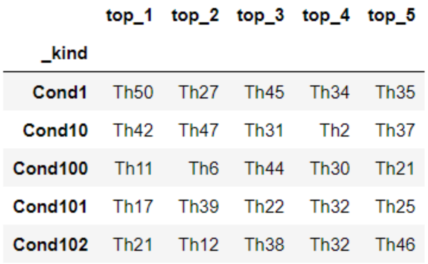
```
           id    _kind _therapy  _successful
0           0  Cond240     Th49           86
1           0  Cond240     Th45           10
2           0  Cond240     Th45          100
3           0   Cond39     Th17          100
4           0  Cond309     Th47          100
...       ...      ...      ...          ...
103170  10933  Cond214     Th22           50
103171  10933  Cond214     Th23           20
103172  10933  Cond214     Th26          100
103173  10935  Cond267     Th48          100
103174  10935   Cond42     Th33          100

[103175 rows x 4 columns]
```

Figure 4: Filtering Dataframe

6.4 Generating top therapy

We have created two dataframe for Conditions and Trials. Then merged both dataframe. After that we have created a condition-therapy interaction matrix based on the success rate and pass the utility matrix to similar neighbor method where the method returns a set of similar therapy respect to the condition ids (Figure 5). After getting the similar neighbors of the therapies we filtered it by the particular condition Id. Thus we get a top therapy list for a specific condition Id. Finally we iterate these set of similar therapy and return a universal top therapy.

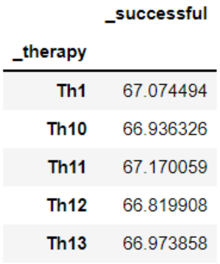


	top_1	top_2	top_3	top_4	top_5
_kind					
Cond1	Th50	Th27	Th45	Th34	Th35
Cond10	Th42	Th47	Th31	Th2	Th37
Cond100	Th11	Th6	Th44	Th30	Th21
Cond101	Th17	Th39	Th22	Th32	Th25
Cond102	Th21	Th12	Th38	Th32	Th46

Figure 5: Finding Similar Neighbor

6.5 Calculating mean success rate for trials

To calculate mean rating of successful trials, initially we performed a groupby operation with therapy and success rate in condition-trial dataframe which is named after "merged data". Then we applied mean operation (Figure 6).



	_successful
_therapy	
Th1	67.074494
Th10	66.936326
Th11	67.170059
Th12	66.819908
Th13	66.973858

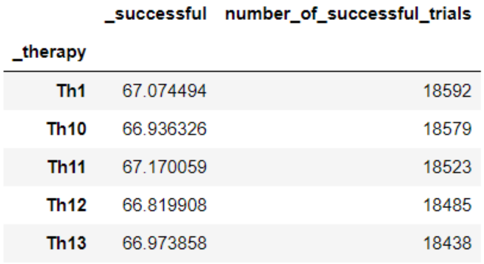
Figure 6: Mean Success Rate for Trials

6.6 Calculating number of successful trials

To calculate total number of successful trials, initially we performed a groupby operation with therapy and success rate in condition-trial dataframe which is named after "therapy success rate". Then we applied count operation (Figure 7).

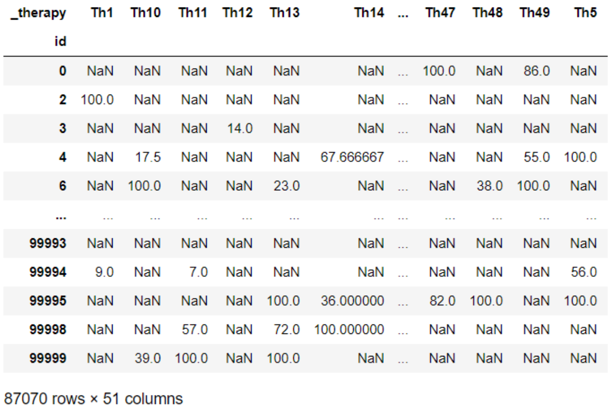
6.7 Generating patient-therapy interaction matrix

We generate a utility matrix indexing by "patient id", followed by the success rate of every therapy. This patient-therapy interaction matrix provides a relationship between individual patient and individual therapy which helps us to get a best therapy ranking list for recommendation (Figure 8).



	_successful	number_of_successful_trials
_therapy		
Th1	67.074494	18592
Th10	66.936326	18579
Th11	67.170059	18523
Th12	66.819908	18485
Th13	66.973858	18438

Figure 7: Number of Successful Trials



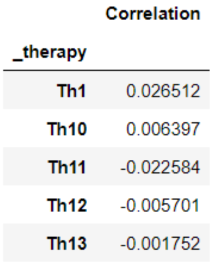
_therapy	Th1	Th10	Th11	Th12	Th13	Th14	...	Th47	Th48	Th49	Th5
id											
0	NaN	NaN	NaN	NaN	NaN	NaN	...	100.0	NaN	86.0	NaN
2	100.0	NaN	NaN	NaN	NaN	NaN	...	NaN	NaN	NaN	NaN
3	NaN	NaN	NaN	14.0	NaN	NaN	...	NaN	NaN	NaN	NaN
4	NaN	17.5	NaN	NaN	NaN	67.666667	...	NaN	NaN	55.0	100.0
6	NaN	100.0	NaN	NaN	23.0	NaN	...	NaN	38.0	100.0	NaN
...	...	...	...	...	...	...	...	...	...	...	...
99993	NaN	NaN	NaN	NaN	NaN	NaN	...	NaN	NaN	NaN	NaN
99994	9.0	NaN	7.0	NaN	NaN	NaN	...	NaN	NaN	NaN	56.0
99995	NaN	NaN	NaN	NaN	100.0	36.000000	...	82.0	100.0	NaN	100.0
99998	NaN	NaN	57.0	NaN	72.0	100.000000	...	NaN	NaN	NaN	NaN
99999	NaN	39.0	100.0	NaN	100.0	NaN	...	NaN	NaN	NaN	NaN

87070 rows × 51 columns

Figure 8: Patient Therapy Interaction Matrix

6.8 Finding correlation with other therapies

We fetched the top therapy before from the "get top therapy" method which is discussed in section-5.4. Now, by this top therapy we get all the ratings from patient-interaction matrix datframe. Then, we apply Pearson's correlation with that to recommend therapy (Figure 9).



	Correlation
_therapy	
Th1	0.026512
Th10	0.006397
Th11	-0.022584
Th12	-0.005701
Th13	-0.001752

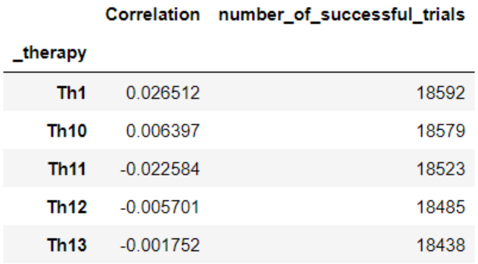
Figure 9: Finding correlation with other therapies

6.9 Joining with number of successful trials

After getting the correlation, we join these correlation values with number of successful trials respect to the therapies (Figure 10).

6.10 The final recommendation

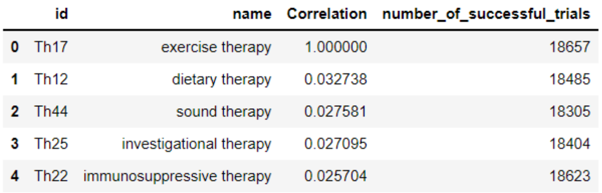
After joining, we sort the dataframe by "correlation" and filter it by "number of successful trials" setting a specific value and order



_therapy	Correlation	number_of_successful_trials
Th1	0.026512	18592
Th10	0.006397	18579
Th11	-0.022584	18523
Th12	-0.005701	18485
Th13	-0.001752	18438

Figure 10: Joining with Number of Successful Trials

the therapies in descending order. We joined the filtered datframe with therapy to get the therapy name. Finally we suggest top 5 therapies those are effective and similar for a particular patient (Figure 11).



	id	name	Correlation	number_of_successful_trials
0	Th17	exercise therapy	1.000000	18657
1	Th12	dietary therapy	0.032738	18485
2	Th44	sound therapy	0.027581	18305
3	Th25	investigational therapy	0.027095	18404
4	Th22	immunosuppressive therapy	0.025704	18623

Figure 11: The Final Recommendation

7 DATASET

To create dataset, we scrapped data from several websites such as Wikipedia, healthcare websites those have some set of information like patient profile, health condition, therapy, and trials. Here, we used python beautiful soup library to extract the data from websites. After extracting the data, we wrote a python script to generate the dataset randomly. The dataset is classified with three main entities patients, conditions and therapies. And the patient entity is segmented with patient condition and trials,

7.1 Conditions

We extracted condition name as well as corresponding condition type from a healthcare website and created a dataset for Conditions generating randomized condition id.

```
"Conditions": [
    {
        "id": "Cond1",
        "name": "Abdominal aortic aneurysm",
        "type": "Mental health"
    },
    .
    .
]
```

7.2 Therapies

We scrapped Therapy name from Wikipedia and created a data set generating randomize therapy id setting with a corresponding therapy type.

```
"Therapies": [
    {
        "id": "Th1",
        "name": "abortive therapy",
        "type": "Systemic Therapy"
    },
    .
    .
]
```

7.3 Patients

We imported a CSV file for patient data from a github repository. We wrote a ptython script to select some basic entities from the csv file for patient dataset. Further, we have added some random values for patient age. We have added some other attributes such as conditions and trials to generate a complete patient dataset.

```
"Patients": [
    {
        "id": 1,
        "name": "Jaco Geldenhuys",
        "gender": "M",
        "age": 80,
        "conditions": [
            {
                "id": "pc1",
                "diagnosed": "20040806",
                "cured": "20090517",
                "kind": "Cond315"
            },
            .
            .
        ],
        "trials": [
            {
                "id": "tr1",
                "start": "20090408",
                "end": "20121128",
                "condition": "pc1",
                "therapy": "Th15",
                "successful": "52"
            },
            .
            .
        ]
    },
    .
    .
]
```

8 EXPERIMENTAL EVALUATION

For evaluation purpose we used Root Mean Square Error - RMSE as a baseline method. To evaluate how efficient our recommendation system algorithm is, we created a dataframe doing merge between two subset conditions and trials of patient dataset. We filtered the dataframe by some specific condition. We selected randomly 5 rows from the given dataset and took the success rate corresponding to the patient condition IDs which is the actual ratings. After that we took the predicted rating from our recommendation algorithm tuning with before selected patients condition IDs. Finally, we calculated RMSE using the actual and corresponding predicted ratings. We repeated the process two

Patient ID	Condition ID	Actual	Predictive	Error
43163	Cond109	42	2.39	39.61
41539	Cond224	100	100	0.0
28855	Cond229	100	2.99	97.01
77625	Cond85	87	2.14	84.86
23621	Cond204	74	1.58	72.42
			RMSE 01	68.44

Table 1: RMSE calculation set-1

Patient ID	Condition ID	Actual	Predictive	Error
95458	Cond282	42	2.19	39.81
32561	Cond287	100	0.48	99.52
88668	Cond97	72	1.86	70.14
72546	Cond231	100	1.41	98.59
15726	Cond168	58	2.30	55.70
			RMSE 02	76.46

Table 2: RMSE calculation set-2

Patient ID	Condition ID	Actual	Predictive	Error
8369	Cond110	15	0.85	14.15
80325	Cond32	56	1.21	54.79
90126	Cond55	45	2.01	42.99
27611	Cond221	58	1.08	56.92
80232	Cond306	65	2.23	62.77
			RMSE 03	49.46

Table 3: RMSE calculation set-3

times to see the difference between several predictions for particular patient conditions.

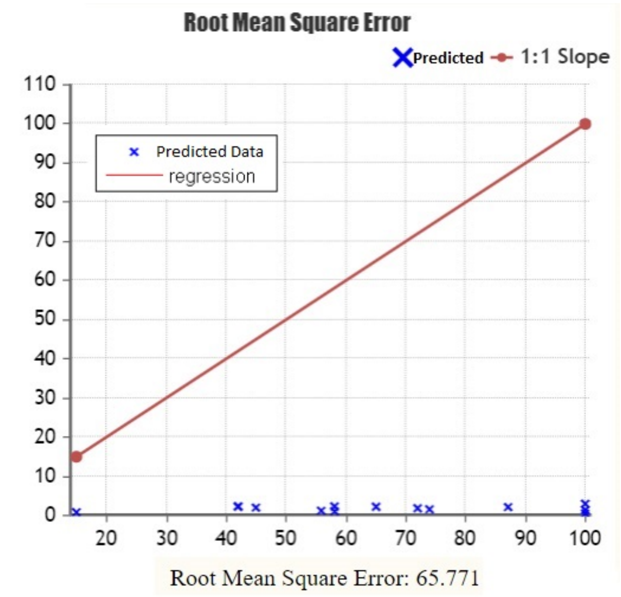


Figure 12: Graphical representation of RMSE

From these above 3 experimental evaluation set of data we calculated the RMSE score is 65.77. We have draw a graph for the analysis of the error. The horizontal line represents the actual ratings and the vertical line represents the predicted ratings. Here we can see a slope which is the regression line and the cross points are the predicted points. The regression line investigates the relationship between actual and predicted values. The distance from the regression line to data points represents the variation of ratings. In our experiment, the RMSE score is not good enough. All the predicted points are far from the regression line that means there is a gap between the predicted ratings and actual ratings (Figure 12).

9 CONCLUSION

Therapy recommendation depends on numerous contributing variables that's why it's always hard to recommend a therapy. A patient requires a personalized treatment based on his health condition, previous medical history, and symptoms. So, a physician need to use his previous knowledge as well as need to analyze patients previous history to suggest a efficient therapy. It is difficult for a doctor to analyze patients previous history and cross match with the patients present symptoms or conditions. That's why there is a chance to have less efficiency in suggested therapy and it would may occur complication for patients. That's why we build a therapy recommended system that will suggest a best therapy by analyzing the patient's conditions, previous trials, and making correlation with other similar successful therapy. We used item based collaborative filtering to recommend therapy and the result is satisfactory. In order to make our recommended system more efficient we have to implement collaborative filtering with clustering. It would make the prediction rate more accurate comparing to the actual success rate.

REFERENCES

[1] Joseph Konstan Badrul Sarwar, George Karypis and John Riedl. 2001. Item-Based Collaborative Filtering Recommendation Algorithms. *ACM.org* (2001). https://dl.acm.org/doi/pdf/10.1145/371920.372071

[2] Claudine Gehin Georges Delhomme Eric McAdams Fabrice Axisa, Pierre Michael Schmitt and André Dittmar. 2005. Flexible Technologies and Smart Clothing for Citizen Medicine, Home Healthcare, and Disease Prevention. *IEEE transactions on information technology in biomedicine* 09 (Sept 2005). https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=1504802

[3] John S. Breese David Heckerman and Carl Kadie. 1998. Empirical Analysis of Predictive Algorithms for Collaborative Filtering. . *In Proceedings of the 14th Conference on Uncertainty in Artificial Intelligence* (1998), 43–52. https://arxiv.org/ftp/arxiv/papers/1301/1301.7363.pdf

[4] Wullianallur Raghupathi and Viju Raghupathi. 2014. Big data analytics in healthcare: promise and potential. *Health Information Science and Systems* (Feb 2014), 1–10. https://link.springer.com/article/10.1186/2047-2501-2-3

[5] J. Ben Schafer. 2005. The Application of Data-Mining to Recommender Systems. 50, 1 (2005), 1–8. http://www.cs.uni.edu/~schafer/publications/dmChapter.pdf

[6] Ray Sipherd. 2018. The third-leading cause of death in US most doctors don't want you to know about. *CNBC.com* (Feb 2018). https://www.cnbc.com/2018/02/22/medical-errors-third-leading-cause-of-death-in-america.html

[7] Zheyun Zhong and Yinsheng Li. 2016. A Recommender System for Healthcare Based on Human-Centric Modeling. *IEEE International Conference on e-Business Engineering* (Nov 2016), 282–286. https://doi.org/10.1109/ICEBE.2016.055