TASK DO: A DAILY TASK RECOMMENDER SYSTEM

A THESIS IN

Computer Science

Presented to the Faculty of the University
of Missouri-Kansas City in partial fulfillment of
the requirements for the degree

MASTER OF SCIENCE

By

NIKHIL SAI SANTOSH GURRAM

BS., Computer Science, University of Missouri – Kansas City, 2018

Kansas City, Missouri

2019

©2019

NIKHIL SAI SANTOSH GURRAM

ALL RIGHTS RESERVED

TASK DO: A DAILY TASK RECOMMENDER SYSTEM

Nikhil Gurram, Candidate for the Master of Science Degree

University of Missouri – Kansas City, 2019

ABSTRACT

Time is a constant entity and an invaluable element for every living person on this planet. Even with all modern-day technologies being available, many individuals like working professionals, students, and house makers often find a lack of time and time management as problems for successful task accomplishment. Many people face challenges in allocating time for their day to day work and personal life activities. One of the key reasons for this failure in task accomplishment is inefficient planning strategies for day to day tasks. There are many task management and to-do-list applications which focus on registering, organizing, sharing, and visualizing tasks, but most of them do not advise on optimal task management and recommendations for better performance. This problem has driven us to contribute a task recommender system which suggests a specific type of tasks to users based on their history of tasks and various factors at that specific time. This system not only suggests a specific type of task for the user but also collects feedback from the user to make the recommender system learn on how to provide useful recommendations thus making the users time much productive. For this system, we have taken some factors into consideration such as Day of the week, Time of the day, Type of the task, Weather, Location and Task completion success percentage. We have designed a rank score algorithm by drilling down to relevant data and by calculating Phi -Correlation

on Task completion success percentage. This algorithm is used to provide recommendations for users for optimal task performance.

Keywords: Recommender Systems, Phi-Correlation coefficient, Task Management

APPROVAL PAGE

The faculty listed below, appointed by the Dean of the School of Computing and Engineering, have examined a thesis titled “TaskDo: A Daily Task Recommender System” presented by Nikhil Sai Santosh Gurram, candidate for the Master of Science degree, and hereby certify that in their opinion, it is worthy of acceptance.

Supervisory Committee

Mohammad Amin Kuhail, Ph.D., Committee Chair
Department of Computer Science Electrical Engineering

Vijay Kumar, Ph.D.
Department of Computer Science Electrical Engineering

Goli Preetham, Ph.D.
Department of Computer Science Electrical Engineering

TABLE OF CONTENTS

ABSTRACT	iii
LIST OF TABLES	vii
LIST OF ILLUSTRATIONS	viii
ACKNOWLEDGEMENTS	ix
CHAPTER 1. INTRODUCTION.....	1
1.1 Motivation	1
1.2 Proposed Solution	2
CHAPTER 2. BACKGROUND AND RELATED WORK.....	4
2.1 Overview.....	4
2.2 Recommendation Systems and Related Work.....	5
2.3 Algorithms and Processes.....	6
CHAPTER 3. TASK DO – A DAILY TASK RECOMMENDER SYSTEM	7
3.1 Design	7
3.2 Methodology	11
CHAPTER 4. EVALUATION	18
4.1 Evaluation Plan	18
4.2 Evaluation Implementation	19
4.3 Evaluation with KNN – Regression model	21
CHAPTER 5. CONCLUSION AND FUTURE WORK.....	23
VITA	27

LIST OF TABLES

Table 1 . Task Type table.....	9
Table 2. Type of the Day	10
Table 3. Time of the Day	10
Table 4. Task Completion Satisfaction Rating Categories.....	11
Table 5 . Sample TaskDB table	12
Table 6. Sample Weekly Plan.....	13
Table 7 . Correlation Table.....	16

LIST OF ILLUSTRATIONS

Figure 3.1: System Architecture Figure.....	7
Figure 3.2: TaskDB Drill-down.....	14
Figure 3.3: Binary Encoding	15
Figure 4.1: Histogram for Random Recommendations vs. TaskDo Recommendations...	19
Figure 4.2: The detailed version of Figure 4 representing two recommendations.....	20
Figure 4.3: Histogram of Rank Score for KNN vs. Task Do recommendations	21
Figure 4.4 The detailed version of Fig.4.3 representing both recommendations.....	22

ACKNOWLEDGEMENTS

Firstly, I would like to thank Dr. Mohammad Amin Kuhail for giving me this opportunity to pursuing research under him. His guidance and direction have been useful for me not only for this research but also for the rest of my professional life. Secondly, I would like to thank Dr. Kumar and the University of Missouri – Kansas City for introducing and providing such quality education and experience to me, respectively. My life has been changed since I came to UMKC.

Finally, I would like to thank my parents, friends, and loved ones for being by my side and for providing immense support and strength to me.

CHAPTER 1. INTRODUCTION

1.1 Motivation

Striving to be productive remains a challenge for many workers, professionals, and students alike. Various researches and surveys state that

- 70% of employees work beyond the scheduled time and on weekends; more than half cited "self-imposed pressure" as the reason. [1]
- The survey, conducted by Greenfield Online[2], found that nearly half of college students (47 percent) feel their high school did not prepare them with the organizational skills required to do well in college. Moreover, 54 percent felt they would get better grades if they "got organized and stayed organized."
- The same survey states that 87 percent of students say that better time management and organization skills would help them get better grades.

Due to this reason, many individuals are facing stress-related problems, and these are the statistics from various stress-related survey sources:

- 71% of white-collar workers feel stressed about the amount of information they must process and act on while doing business; 60% feel overwhelmed.[3]

To recover from these sorts of problems, spending time organizing daily tasks would be helpful. Research says that:

- For every hour of planning, 3 to 4 hours are saved from redundancy, waiting for information, not being prepared and poorly managed tasks [4]
- 10-12 minutes invested in planning your day will save at least 2 hours of wasted time and effort throughout the day [5]

With new age advancements in information technology, we have many applications which focus on time and task management. Task and time management tools such as Todoist [6] and Wunderlist [7] allow users to add, track, organize, manage, and share tasks. However, these tools do not have a mechanism for suggesting tasks to a specific user to be done at a specific time for his/her optimal performance over time. Hence, users are left on their own to plan their daily schedule. Further, these tools have implemented some visual analytics and data visualization techniques to help users to understand their productivity and how it changes over time.

1.2 Proposed Solution

Driven by the importance of time and task management, and lack of the tools which suggest a specific type of task for a specific date and time, we are inspired by the research question: “Based on the user’s history of completed tasks, can we recommend certain task types to be done at certain days of The week and times of day to increase a user’s productivity?”. To answer this question, we contribute TaskDo, a task recommender system which suggests a specific task to be done on certain days of the week and at certain times of the day. In generating the recommendations, the system relies upon the history of a user’s task completion. For example, since the system has observed that Adam tends to complete his chores at a shorter time on weekends in the morning, the system will recommend to Adam that he does his chores in the morning on weekends. This is a simple case, but it illustrates the point of the system learning about user’s habits of task completion and suggesting what is believed to best the best time

for a specific type of task. Our initial thought is that the recommendations of task types will be affected by many variables such as day of the week, time of day, whether the task is done indoors or outdoors, whether the task is intellectual or physical, and so on.

CHAPTER 2. BACKGROUND AND RELATED WORK

2.1 Overview

There are many task management and calendar applications on the internet and application markets open for use. However, these tools do not recommend users suitable type of tasks for a specific day and time. Timeful [8] was an application that aimed at understanding users' habits and schedules by asking users how often and when they want to do things. The system then assists users with planning their schedule. The idea seems promising, but the application is currently unavailable, which makes it hard to evaluate the accuracy of the application. Users often find it difficult to prioritize or decide upon numerous types of tasks that they might have in their day to day life. Personalized recommendations are gaining popularity in such confusing scenarios to target users' interest and preferences. Many well-known organizations are providing personalized recommendations to their users based on the user history and preferences data they might have already.

Netflix [9], a favorite movie and tv series viewing platform, gives personalized recommendations to the users for their movies or tv series that user might like with a match score percentage. Amazon [10], a leading e-commerce website provides personalized product recommendations to the user based on users' previous buying behavior. These organizations provide recommendations based on Recommendation Systems. Different types of Recommendation Systems and their applications are explained in the section below.

2.2 Recommendation Systems and Related Work

Recommendation Systems used in most of the applications are classified broadly into two different types:

1. Collaborative Filtering
2. Content-Based Filtering

The generic definitions of these Recommendation Systems are as follows:

- Collaborative Filtering:

Collaborative filtering recommends items by identifying other users with a similar taste.

- Content-Based Filtering:

Content-based filtering recommends items based on the user profile; it does not consider other users. [11]

Martínez, A. B., Arias, J. P., Vilas, A. F., Duque, J. G., & Nores, M. L has designed a system which recommends T.V shows for the users using both contents based and collaborative filtering. They have implemented a hybrid model using both the techniques because to avoid the cold start problem (no recommendations at the start) which is caused by collaborative filtering and to avoid the over-specialization problem (showing only very few items from the user profile) caused by content-based filtering. [12]

Bagher, R. C., Hassanpour, H., & Mashayekhi, H have defined a model to estimate user personal interests by using collaborative filtering. They have implemented profile matching and latent factors as two main approaches for modeling the user. They have

built this using a Bayesian non-parametric model, which provides a framework for constructing an evolutionary model. [13]

2.3 Algorithms and Processes

A) Content-Based Filtering

TaskDo relies on the user's previous task history, and the recommendations are generated based on the user's previous task history. It does not depend on the task history of the other users. So, TaskDo is a Content-Based Filtering Recommendation System.

B) Phi Correlation

Phi Correlation or Phi Coefficient is used to measure the correlation between two or more binary variables. Many categorical features used in TaskDo are converted into binary variables to perform the analysis. The correlation coefficient is calculated between these features using Phi-Correlation.

C) Clustering

Clustering is a mechanism used to group the data in such a way that observations in each group are statistically identical or closer to each other. Clustering can be performed either automatically or manually depending on the use case and familiarity of observations. Task Do uses manual clustering of the observations concerning features such as Day of the week, Time of the day, etc. Methodology section further explains the specific process of clustering used in TaskDo.

CHAPTER 3. TASK DO – A DAILY TASK RECOMMENDER SYSTEM

3.1 Design

TaskDo provides task recommendations for a specific user for a specific day of the week and a specific time of the day. These recommendations are provided based on the users' previous task history and task performance. Several dependent and independent modules have been identified to develop such a task recommender system. The design of TaskDo consists of the following subsystems shown in figure 1.

Figure 3.1: System Architecture Figure

Figure 3.1 shows the system architecture for TaskDo, task recommender system. Initially, users provide their daily tasks and related variables such as Day of the week, Time of the Day, Type of the Day, etc. (explained in detail in the next section) into the system. Later, after the completion of the task, the system will ask the user to record feedback on task completion. This process will be continued for a predefined amount of time to avoid the cold start problem.

Eventually, after recording many numbers of tasks for a specific period, the recommender system starts giving Task Recommendations. For example, if the user records the successful completion of chores on Monday early mornings frequently, and if the same user fails to complete other types of tasks on Monday early mornings successfully, the recommender system recommends the user to perform chores over other task types on a Monday early morning to increase productivity at that specific time slot. Once when the user receives recommendations, the recommender system once again collects the task completion feedback from the users to check whether the provided recommendations were helpful or not. The system provides the recommendations once again based upon the updated feedback provided by the user, Hence following a cyclic procedure.

a) Task Preferences and user feedback

During the process of building this recommender system, we recognized that there a huge variety of user tasks, which needed a categorization or segregation to be able to provide recommendations. So, we listed down the tasks and narrowed their scope into several different categories based on the nature of the task. The list of Task Types, according to nature and relevant task examples, are illustrated in Table 1.

Table 1. Task Type table

Task Example	Task Type
Studying, Reading, Homework, etc.	Intellectual
Prayers, Meditation, etc.	Spiritual
Swimming, Jogging, Workout, etc.	Fitness & Health
Kids, Parents, Friends, etc.	Social
Groceries, Sending Mails, Shopping, etc.	Errands
Cooking, Cleaning, Laundry, etc.	Chores

Later, we discovered that there are some dependent variables such as Type of the Day and Time of the Day illustrated in Table 2 and Table 3, respectively, which can influence the user's productivity. When a user enters his task into the system, he is also required to enter all the subcategories for the dependent variables regarding that specific task, as explained in Table 1, 2, 3.

Table 2. Type of the Day

Type of the Day
Sunday
Monday
Tuesday
Wednesday
Thursday
Friday
Saturday

Table 3. Time of the Day

Time of the Day	From(hh:mm:ss) - To(hh:mm:ss)
Early Morning	04:00:00 – 07:59:59
Morning	08:00:00 – 11:59:59
Afternoon	12:00:00 – 15:59:59
Evening	16:00:00 – 19:59:59
Night	20:00:00 – 23:59:59
Mid-Night	00:00:00 – 03:59:59

After completion of the task (or the completion of allocated time for the task), the system prompts the user and collects Task Completion Satisfaction Rating from the user regarding task completion. The Task Completion Satisfaction Rating is collected

from the user in terms of Categorical 1 to 5 rating system. Table 4 is the illustration of meanings for each category of Task Completion Satisfaction Rating

Table 4. Task Completion Satisfaction Rating Categories

Task Completion Satisfaction Rating	Meaning
1	Very Low Task Completion Satisfaction
2	Low Task Completion Satisfaction
3	Medium Task Completion Satisfaction
4	High Task Completion Satisfaction
5	Very High Task Completion Satisfaction

Eventually, after some amount of time each user has their Database in the system (Task DB) which contains all his/her tasks with feedback and the recommender system provides recommendations based on the tasks and feedback recorded in Task DB, hence making it a Content-Based Recommender System. The problem that we faced in developing a recommender system is that we do not have the task DB of specific user pre-recorded with us to develop and evaluate the recommendations. For this reason, we simulated a thousand datasets such that a permutation and combination of each task type with each day, possible day type.

3.2 Methodology

Whenever a user enters a task, its dependent variables, and task completion satisfaction rating into the recommender system, it gets stored into the Task DB as input

similar to a row in Table 2. For example, If a user performs Intellectual task on a Monday early morning, and if the same user records Very High Task Completion Satisfaction rating, it gets entered into TaskDB as the first row in Table 5.

Table 5. Sample TaskDB table

Time of the day	Type of the Day	Task Category	Task completion Satisfaction Rating
Early Morning	Monday	Intellectual	5
Afternoon	Tuesday	Chores	3
Evening	Sunday	Spiritual	4

Since a User Interface design is not available at the initial start, we have decided to give the task recommendations as a Weekly Plan for the user. After recording all tasks into the system in a way mentioned above, the two most successful type of tasks are provided as recommendations concerning the type of the day and time of the day as a weekly plan. Table 6 is the illustration of sample Weekly Plan that we want to provide as task recommendations. Each row represents recommendations for each day of the week. Each column represents recommendations for each Time of the Day.

Table 6. Sample Weekly Plan

	Early Morning	Morning	Afternoon	Evening	Night	Midnight
Saturday	1.fitness 2.errands	1. 2. chores	1. 2. social	1. 2. Spiritual	1. Spiritual 2. fitness	1. chores 2.Spiritual
Sunday	1. 2.Spiritual	1.Spiritual 2. fitness	1.Spiritual 2. intellectual	1.intellectual 2.	1. chores 2.intellectual	1.errands 2. fitness
Monday	1. 2. fitness	1. social 2.	1. social 2. intellectual	1. social 2.	1. social 2. fitness	1. chores 2. Spiritual
Tuesday	1. 2. social	1. fitness 2.	1. chores 2.	1. errands 2.	1. Spiritual 2. fitness	1. chores 2. errands
Wednesday	1. social 2. fitness	1. Spiritual 2. social	1. 2. social	1. 2. chores	1. 2. intellectual	1. 2. errands
Thursday	1. social 2.	1. Spiritual 2.	1. fitness 2.	1. fitness 2. Spiritual	1. fitness 2. social	1. errands 2. intellectual
Friday	1. chores 2. Spiritual	1. 2. errands	1. chores 2. fitness	1. 2. fitness	1. fitness 2. Spiritual	1. social 2. fitness

To generate these type of weekly plan recommendations shown in Table 3, and to provide the custom recommendations for each Day of the week and each time of the day, Task DB is drilled down to a specific day of the week and specific time of the day to create the custom recommendations. For example, let's assume that the system needs to recommend the two most successful tasks for Saturday as Day of the week and Early morning as the time of the day. The system first drills down the whole user Task DB into

the data portion with having Saturdays data as the first step, and it later drills down this portion to only data that belongs to Early Morning. Figure 2 explains this entire stepwise Drill-down process.

Figure 3.2: TaskDB Drill-down

After drilling down data as shown in Figure 3.2 according to day and the time of the day, internally in the recommender system, we encode the values of Task Category and Task Completion Satisfaction Rating into binary categorical variables as shown in Figure 3.3

Time of the day	Type of the Day	Task Category	Task Completion Satisfaction Rating
Early Morning	Saturday	Intellectual	5

Task Category							Task completion Feedback				
Intellectual	Physical	Spiritual	Fitness	Social	Errands	Chores	Very low	Low	Medium	High	Very High
1	0	0	0	0	0	0	0	0	0	0	1

Figure 3.3. Binary Encoding

Once the system does the binary encoding as shown in figure 3, it calculates the correlation between the Task types (Intellectual, Physical, etc.) and Task Completion Satisfaction Rating categories (1, 2, 3, etc.). For example, the correlation between Intellectual task type and very low Task Completion Satisfaction Rating, Intellectual task type, and low Task Completion Satisfaction Rating, etc. for all task types gets calculated. For this correlation analysis, we used the Phi correlation coefficient to calculate the correlation coefficient between both of these binary variables. After performing the correlation analysis, the correlation table looks like Table 7.

Table 7. Correlation Table

	Intellectual	Chores	Social	Errands	Fitness	Spiritual
Very Low	0.915767	0.418168	-0.4745	0.893217	0.17677	0.015298
Low	0.437112	-0.32118	0.476863	0.587735	0.92127	-0.58766
Medium	-0.76901	0.752388	-0.38813	0.55597	0.999404	0.61027
High	0.902603	0.050775	0.448632	0.281429	-0.13176	0.037248
Very High	0.960326	0.685226	-0.9797	-0.15146	0.303423	0.078313

So, the final recommendations into the weekly plan get provided by calculating the rank score of each task type with the formula, as shown below:

$$\begin{aligned} \text{Rank Score for each task type} = & 2 * (\text{Correlation of very high Task Completion Satisfaction} \\ & \text{Rating vs. Task Type}) + 1 * (\text{Correlation of high Task Completion Satisfaction Rating vs.} \\ & \text{Task Type}) + 0 * (\text{correlation of medium Task Completion Satisfaction Rating vs. Task} \\ & \text{Type}) - 1 * (\text{Correlation of low Task Completion Satisfaction Rating vs. Task Type}) - \\ & 2 * (\text{Correlation of very low Task Completion Satisfaction Rating vs. Task Type}) \end{aligned}$$

The reason for assigning weights -2 to 2 for different correlation types in the rank score formula is because, we don't want to provide any task types which have evidence of low/very low performance as recommendations when compared to high/very high performance. After calculating the rank score for each task type, the two highest positively rank scored task types gets provided as recommendations into a weekly plan for every type of the day and time of the day combination as shown in Table 6. The reason for only recommending positively ranked task types is because if a task type is negatively ranked, it means that the task type shows very low/low task

satisfaction rating and we don't want to provide any task type which has very low/low task satisfaction rating. The pseudocode for this entire process is as written below;

Import Task DB

For all Type of the day values:

For all Time of the day values:

- *Drill down TaskDB as shown in figure2*
- *Apply binary encoding to all values in the Drilled Task DB*
- *Calculate the Phi correlation between the Task type categories and Task Completion Satisfaction Rating categories*
- *Calculate a rank score for each task types with the formula given above*
- *Return two best task types with the highest positive rank score into the weekly plan for selected Type of the day and Time of the day*

CHAPTER 4. EVALUATION

4.1 Evaluation Plan

As explained in the beginning, the purpose of this recommender system is to recommend a particular type of task to increase users optimal performance. Without any custom task recommendations, any person is likely to pick up and do the tasks randomly. So, we have created about a thousand random datasets with each type of task and the type of day permutations and combinations and task satisfaction score with random seeds. The pseudo code for generating random data sets is below:

In the Range of iteration count equal to 1 to 1000:

The seed is equal to the iteration count

For all Type of the day values:

For all Time of the day values:

Pick any two types of tasks randomly as recommendations

Calculate the rank scores for random recommendations

Save it as one of the thousand random datasets with random recommendations

Then we calculated the recommendations for random datasets with TaskDo, as explained in section 3.2 and calculated the rank scores for them. Then we plotted the rank scores of random recommendations vs. the TaskDo recommendations for all of the datasets on a histogram with scale -2 to 2. The reason that we have used this scale is that -2 and 2 are the possible maximum and minimum values of the Rank Score formula mentioned in section 3.2. When a task type completely correlates to a very high completion satisfaction rating, it yields a score of 2 in rank score formula. Also, When a

task type completely correlates to a very low completion satisfaction rating, it yields a score of -2 in rank score formula. Hence, -2 and 2 are the minimum and maximum values of the rank score, respectively.

4.2 Evaluation Implementation

Figure 4.1 represents the histogram, which shows the difference between Random Recommendations vs. TaskDo Recommendations.

Figure 4.1: Histogram for Random Recommendations vs. TaskDo Recommendations

The orange line in Figure 4 represents Random Recommendations, and the blue line represents TaskDo Recommendations. We can infer from the graph that Random Recommendations are widespread over -2 to 2 referring to recommendations which have a negative and positive correlation with task satisfaction, whereas TaskDo

Recommendations only has positively correlated recommendations which yield to higher productivity for its users.

Figure 4.2: The detailed version of Figure 4.1 representing two recommendations

Figure 4.2 represents a detailed version of Histogram in Figure 4 including the top two recommendations provided for each cell in the weekly plan as shown in Table 6 (each cell contains recommendations for one type of the day and one time of the day). The orange and red lines represent Random Recommendations, whereas the blue and pink lines represent TaskDo Recommendations. We can once again infer with this graph that Random Recommendations are coinciding with each other over scale of -2 to 2 whereas TaskDo Recommendations only provide positively correlated task types yielding to higher productivity for its users.

4.3 Evaluation with KNN – Regression model

One of the very popular content-based recommendation algorithm is KNN – Regressor (K-Nearest Neighbor Regressor model). The basic functionality of this model is to predict a score of an unknown observation by taking a statistical measure such as an average into account for the observations in the training set which are at a closer distance to the unknown observation. We have applied the same model to the randomly generated datasets in generating recommendations to see how it performs in comparison with TaskDO. Figure 4.3 and Figure 4.4 are the illustrations of histograms which show rank scores of KNN regressor recommendations vs. TaskDo recommendations.

KNN vs. TaskDo

Figure 4.3 Histogram of Rank Score for KNN vs. Task Do recommendations

In the figure 4.3 we can see that both of the generated recommendations are almost similar in terms of score but TaskDo slightly out performs KNN by not recommending the negative values and by having high average score that is recorded by recommendations. Figure 4.4 is the detailed version of figure 4.3 as illustrated below.

KNN Vs. TaskDo Detailed

Figure 4.4 The detailed version of Fig.4.3 representing both recommendations

We can also see in Figure 4.4 that TaskDo outperforms a little bit when compared to KNN regressor in both of the recommendations.

CHAPTER 5. CONCLUSION AND FUTURE WORK

- Since the data that we are using currently is simulated by ourselves to perform validation over the developed system, the system may not be the best fit yet with real-world data. For this reason, we want to anonymously collect data from 20 different volunteers for over two months to get a real-world task dataset. After getting this dataset, we want to perform the same correlation analysis and provide recommendations back to the user regarding the weekly plan and check whether the users are satisfied with the provided weekly plan. If not, we want to check whether any adjustments can be made to the system based on the user's feedback on provided recommendations.
- We also want to include the importance and impact of strict deadlines for user tasks and make the system smart enough to provide recommendations based on the strict deadlines and timelines that user needs to follow while improving their task productivity on the other hand.
- Also, there are many other dependent variables like Weather, Task Duration, and Task Location, which may also impact the user's productivity in doing tasks. So, we want to make the system smarter to include those variables and provide recommendations considering them as well.
- Moreover, since most the data that we are using to provide recommendations is categorical, we also want to use Machine Learning classification algorithms such as Bayesian classifiers, cluster analysis, decision trees, and Artificial Neural Networks to provide better task recommendations to the user.

- The idea and behavior of tasks done by people on weekdays/weekends are different when compared to different countries. So, we want to take that into account while we expand our system to work with such data and people's behaviors
- No matter how well the recommendations perform and how good the recommender engine is there will always be extreme cases and task behaviors from people which may need additional accommodations. So, we want to keep it in mind while we expand our system to further level.

REFERENCES

- [1] Key Organization Systems. "Time management statistics." Internet:
<http://www.keyorganization.com/time-management-statistics.php>. [July 31, 2019].
- [2] ReliablePlant, "College students struggle with organizational skills," Internet:
<https://www.reliableplant.com/Read/3429/college-students-struggle-with-organizational-skills>. [Jul. 31, 2019].
- [3] A. M. Moore, "Organizing Statistics." Internet:
<http://mooreorganization.com/stats.html>. [Jul. 31, 2019].
- [4] S. Borsheim. "Organizing & time management statistics." Internet:
<https://www.simplyproductive.com/2012/03/time-management-statistics>,
March 12, 2012 [July 31, 2019].
- [5] B. Tracy, *Eat That Frog!: 21 Great Ways to Stop Procrastinating and Get More Done in Less Time*. San Francisco, CA: Berrett-Koehler Publishers, Incorporated, 2001.
- [6] Todoist. "The best to do list app & task manager." Internet:
<https://todoist.com/?lang=en>. [July 31, 2019].
- [7] Wunderlist. "To-do list, reminders, errands - App of the year!" Internet:
<https://www.wunderlist.com/>. [Jul. 31, 2019].
- [8] D. Etherington, "Google Acquires Timeful To Bring Smart Scheduling To Google Apps," *TechCrunch*, 04-May-2015. Internet:

- <https://techcrunch.com/2015/05/04/google-acquires-timeful-to-bring-smart-scheduling-to-google-apps/>. [Jul. 31, 2019].
- [9] Netflix. "See what's next." Internet: <https://www.netflix.com/>. [31-Jul-2019].
- [10] Amazon. "Online shopping for electronics, apparel, computers, books, DVDs & more,". Internet: <https://www.amazon.com/>. [Jul. 31, 2019].
- [11] F. Isinkaye, Y. Folajimi, and B. Ojokoh, "Recommendation systems: Principles, methods and evaluation," *Egyptian Informatics Journal*, vol. 16, no. 3, pp. 261–273, 2015.
- [12] A. B. B. Martinez, J. J. P. Arias, A. F. Vilas, J. G. Duque, and M. L. Nores, "Whats on tv tonight? An efficient and effective personalized recommender system of TV programs," *2009 Digest of Technical Papers International Conference on Consumer Electronics*, 2009. Internet: <https://doi.ieeecomputersociety.org/10.1109/ICCE.2009.5012204>.
- [13] R. C. Bagher, H. Hassanpour, and H. Mashayekhi, "User trends modeling for a content-based recommender system," *Expert Systems with Applications*, vol. 87, pp. 209–219, 2017.

VITA

Nikhil Sai Santosh Gurram is a transfer student from Vellore Institute of Technology, India to UMKC and he completed his Bachelor's and Master's Degrees in Computer Science and Data Science respectively on a 3+1+1 program. He also worked as a Software Maintenance Intern at PriceSpider, Kansas City from June 2018 and he got promoted as a Big Data Analyst at the same company in January 2019. He wants to see himself as a Data Scientist in the next 2-3 years.