A computational approach to assist healthcare professionals in selecting antibacterial drugs to treat bacterial infections

submitted by

SAYANTIKA CHATTERJEE

in partial fulfilment of the requirements
for the award of the degree of

MASTER OF TECHNOLOGY

ELECTRONICS AND COMMUNICATION ENGINEERING (MACHINE LEARNING)
INDRAPRASTHA INSTITUTE OF INFORMATION TECHNOLOGY DELHI

NEW DELHI- 110020

May, 2023

THESIS CERTIFICATE

This is to certify that the thesis titled **A computational approach to assist healthcare professionals in selecting antibacterial drugs to treat bacterial infections**, submitted by **Sayantika Chatterjee**, to the Indraprastha Institute of Information Technology, Delhi(**IIITD**), for the award of the degree of **Masters in Electronics And Communication(Machine Learning)**, is a bona fide record of the research work done by her under my supervision. The contents of this thesis, in full or in parts, have not been submitted to any other Institute or University for the award of any degree or diploma.

Dr. Angshul Majumdar
Thesis Supervisor
Associate Professor
Dept. of Electronics and Communication
IIIT Delhi, 110020

Place: New Delhi

Date: 20th May 2023

ACKNOWLEDGEMENTS

This study and the research would not have been possible without the invaluable support of my supervisor, Dr. Angshul Majumdar. His guidance, mentorship throughout my research journey helped me a lot to improve the quality of my work. He made great efforts in supporting me in all possible ways. I also like to thank my co-guide Aanchal Mongia for providing insight and expertise that greatly helped the research. I am also grateful to Abhishek Haldar, Neetesh Pandey and Prateek Paul for their assistance. I would also thank my family for encouraging me to pursue my academic goal.Their unwavering support and understanding have helped me to stay motivated and focused during difficult times. I am also grateful to my friends who always motivated me to reach my goals. This section can not be completed without a vote of thanks to academic department for their help and endless support .

ABSTRACT

The reproducibility of experiments has been a long-standing obstruction for farther scientific evolution. Computational methods are being involved to accelerate and to economize drug discovery and the development process. In this work several computational models using matrix completion techniques including matrix factorization, deep matrix factorization, binary matrix completion and graph regularised techniques (graph regularised deep matrix factorisation, graph regularised matrix factorization, graph regularised binary matrix completion and graph regularised matrix completion) have been proposed to predict bacteria-drug association. Here drug-bacteria association matrix is formed. Along with it we gather similarity information using the chemical structure of drugs and genome-genome distance calculator Meier-Kolthoff *et al.* (2022) for bacteria. Using several matrix completion tools, the bacteria-drug association data and similarity data, the present study predicts the set of best possible drugs corresponding to each bacteria in the database. The graph regularised techniques consider the drug-bacteria association matrix along with the similarity information for prediction. To evaluate robustness of the model, cross validation settings on different scenarios have been adopted on the training data. The AUC-AUPR metric is being reported corresponding these scenarios and association between drug-bacteria is being predicted with the help of various graph and non graph regularised methods. The result produced by graph regularised methods are better compared to non graph regularised methods. Hence it can be concluded that the graph regularised methods predicts the association data well. We anticipate that this work will provide opportunities to develop drugs for newly discovered bacteria and, conversely, enable the identification of potential bacteria targets for existing drugs

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ABBREVIATIONS

iiitd	Indraprastha Institute of Information Technology, Delhi
MC	Matrix Completion
MF	Matrix Factorization
BMC	Binary Matrix Completion
DMF	Deep Matrix Factorization
GRMC	Graph Regularised Matrix Completion
GRMF	Graph Regularised Matrix Factorization
GRDMF	Graph Regularised Deep Matrix Factorization
GRBMC	Graph Regularised Binary Matrix Completion
GGDC	Genome Genome Distance Calculator
CV	10-fold Cross Validation in different scenario

CHAPTER 1

INTRODUCTION

Recognising drug target and drug-bacterial interactions are one of the most essential steps in the area of drug-discovery. Successful computational drug discovery techniques are proved to be an effective blueprint for economising and stimulating drug discovery and development process. These computational methods can also reduce time required by the experimental methods. In the year 2020 we witnessed a severe situation where a huge number of people got died due to COVID-19 Zhou *et al.* (2020); Dotolo *et al.* (2021). The disease causes dreadful respiratory syndrome coronavirus 2 which was being outspread across more than 100 countries. This epidemic vandalized people's life, global economy heavily from all perspectives and mostly the pillar of the human race. In this circumstance the analysis involved to develop a new drug is Laborious and require several phases of considerable trials. Hence to deal with this type of scenario the finest way is to repositioning existing drugs. This is no doubt a well-known technique where the existing drugs (which have already been endorsed in the market for release) are analysed or examined for a new disease Ashburn and Thor (2004). Drug-repositioning is generally cost-effective and requires less amount of time compare to introducing new drug as its consequences are well examined. The COVID-19 Zhou *et al.* (2020); Dotolo *et al.* (2021)pandemic brought drug repurposing research on viral infections at the forefront. However, we could not find any methodical study on drug repositioning for bacterial infections. According to a recent estimate 7.7 million people die in the world every year from bacterial infections[1]. This motivates our current work. In this study, we curate a database for drug-bacteria-association and benchmark state-of-the-art algorithms. Practically, this type of techniques could also help clinicians in developing effective treatments for speedily mutating bacteria by pruning the anti-bacteria drug space Ezzat *et al.* (2019). In this work a computational approach is adopted which takes into consideration the genomic distance of the bacteria or their similarity to decide the drug for the bacteria. Hence for any newly introduced bacteria this computational

[1] `https://www.reactgroup.org/news-and-views/news-and-opinions/year-2022/7-7-million-people-die-from-bacterial-infections-every-year/`

method can also determine the corresponding drug by calculating the genome distance or similarity between the new one and the existing bacteria which would be helpful to decide the treatment of bacterial infections. In pursuit of this objective we have formed bacteria-drug association matrix in conjunction with the similarity measures affiliated with drugs (chemical structure similarity) and bacteria (genome-genome distance) and follow a machine learning based approach. There can be various procedures like neighborhood models,feature-based classification models, matrix completion models etc to predict bacteria drug association. A recent experimental study on firmly established drug-target interaction databases shows superlative prediction performance by matrix completion models Ezzat *et al.* (2019). In computer science matrix completion method is used in recommendation systems. The general problem of drug-Infection can be considered as a recommendation system, where drugs are being recommended to treat Bacterial Infections. Here also we deploy matrix completion techniques followed by matrix factorisation, deep matrix factorisation and graph regularised techniques like graph regularised matrix completion, graph regularised matrix factorization, graph regularised deep matrix factorization and graph regularised binary matrix completion to anticipate drug-Infection associations. To validate the robustness of the algorithms we perform 10-fold cross validation in training data on several scenarios by masking or hiding few entries(10% associations in scenario1,10% bacteria in scenario2,10%drugs in scenario3 selected randomly) in the association matrix and predict the corresponding association using several graph and non graph regularised algorithms. We believe there are three ways our work can benefit researchers and clinicians working in this area. First, it can help in identifying usage of existing antibiotics for known bacterial infections. Second, it can help in repurposing existing antibiotics for new bacteria. Third, it can help find potential existing bacterial infections that benefit from a newly developed antibiotic. Of these, we assume that the second point is of prime importance. This is because of the rise of antimicrobial resistance Dadgostar (2019); Murray *et al.* (2022).Over time, through mutations and selection bacteria develops defenses against antibiotics. In such cases, the clinicians need to find newer ways of treatment Schrader *et al.* (2020).We believe that clinicians may use our approach to screen such treatment plans.

CHAPTER 2

Methods

2.1 Dataset Definition

The information about the bacterial infections and the corresponding effective drugs are taken from the antibiotic guidelines 2018 of Christian Medical College, Vellore[1] and the antibiotic policy manual of AIIMS(all india institute of medical sciences),jodhpur, rajasthan[2]. In these two manuals bacterial infections including the causative organisms(bacteria) and the effective drugs or antibiotic are specified.

As mentioned in the introduction, we would be using matrix completion based approach for drug-bacteria-association. In this approach the antibiotics are along the rows and the bacteria along the columns (or vice versa). From the aforementioned guideline, wherever a drug is known to be effective against a bacteria, we have marked it as 1 in the matrix. These are the true positives. Ideally the drugs that are known to be ineffective against a certain bacterial infection should have been marked as the true negative. Unfortunately, this information is not available in the guideline. This problem is not unique to our dataset, it must be noted that almost all biological interaction problems such as drug-target-interaction Shi and Yiu (2015),drug-disease-interaction Gottlieb *et al.* (2011); Luo *et al.* (2016),drug-drug-interaction Wishart *et al.* (2018),drug-virus-interaction Mongia *et al.* (2021)etc.The ensuing matrix is of size 53 X 61; there are 53 antibiotics and 61 bacteria.

This DBA (drug bacteria association) database is useful to analyse and propose antibacterial drug for bacteria. In conjunction with this it may also be utilised to computationally recognise bacteria that a recently identified drug might aim at. The correlated metadata (information about the drugs and the corresponding bacteria) may also assist the clinicians to analyse manually and to have a deeper insight.

[1] https://github.com/sayantika21175/Thesis_2023/blob/main/Antibiotic%20guidelines%20for%20adults%202018-%20CMC%20vellore.pdf

[2] https://www.aiimsjodhpur.edu.in/quick%20docs/Antibiotic%20Policy%20AIIMS%20JDH%202018.pdf

As mentioned before, we will be using graphical matrix completion approaches in this work. This takes into account associated metadata for the drugs and the bacteria. All the associated information on drugs is obtained in the DrugBank vocabulary[3]. From here the unique DrugBank Id for the antibiotics are obtained. However, we do not need the raw data for the drugs but the structural similarity between the drugs. The similarity is computed via the SIMCOMP score Hattori *et al.* (2010). For computing this score, one needs to access the Kyoto Encyclopedia of Genes and Genomes (KEGG). Kanehisa *et al.* (2006) The KEGG Id (kid) is the unique identifier for each KEGG object. The linkage between KEGG and DrugBank is available at the linking file[4]; Wishart *et al.* (2006) In case the drugs were missing in the file it was manually appended. This linked each antibiotic with a DrugBank ID to a corresponding KEGG ID on the KEGG Compound/KEGG Drug database[5] of the KEGG.

For bacterial information Full Genome sequence (nucleotide sequences in FASTA format) is considered. Genome sequence helps in determining the working of the genomes as a whole. It also gives an insight into the biology of many bacterial pathogens and also identify novel antibiotic target. This sequence is fetched from the NCBI (National Center for Biotechnology Information) genome database [6] National Center for Biotechnology Information (Accessed May 5, 2023). This NCBI genome database also contains protein sequence, genome annotation in GFF, GenBank format, Genome Blast, all the genome list for particular species. Each bacteria has its own full genome sequence. Hence, we have total 61 genome sequences. These genome sequences of bacteria are used as an input to calculate similarity among bacteria.

2.2 Similarity Computation

Here analogy among drugs and analogy among the bacteria are calculated with the chemical information of drugs and genome sequences of bacteria respectively. These similarities are useful to design this recommendation system mentioned in this work.

[3] https://www.drugbank.ca/releases/latest#open-data

[4] https://www.drugbank.ca/releases/latest#external-links

[5] https://www.genome.jp/kegg/compound/ and https://www.genome.jp/kegg/drug/

[6] https://www.ncbi.nlm.nih.gov/genome/

2.2.1 Similarity calculation among Drugs

DrugBank IDs of the drugs are there along with the corresponding KEGG IDs. These DrugBank IDs and the corresponding KEGG Compound IDs are mapped together to form a new dataset containing DrugBank ID, KEGG compound ID, KEGG Drug ID and the Drug Name. The drugs either have KEGG compound ID or KEGG Drug ID corresponding DrugBank ID. To compute the similarity among the drugs their chemical significances or chemical structure is considered. There is SIMCOMP (SIMilar COMPound) server for the chemical similarity search. Chemical structure similarity between two drugs is calculated with the help of SIMCOMP score Hattori *et al.* (2010). It is a graph-based method which is based upon the maximum common substructures between the chemical structure of the compounds using the KEGG API page at GenomeNet [7]. To calculate SIMCOMP score cutoff is set as **0.001**. The drugs with no KEGG IDs available and the drugs for which the SIMCOMP score was smaller than the predefined threshold (0.001)are being allotted a similarity score of 0 to other drugs in the dataset and 1 to themselves.

2.2.2 Similarity calculation among Bacteria

Genome-Genome distance calculator[8] Meier-Kolthoff *et al.* (2022) is used to find out the similarity in between two bacteria using their genome sequences. It is a futuristic in silico method to compare genomes of various organisms, thus truly imitating the conventional DDH Meier-Kolthoff *et al.* (2013), except for its pitfalls. GGDC (Genome-Genome distance calculator) Meier-Kolthoff *et al.* (2022) works much faster compare to other techniques. We have also tried Stretcher (Emboss) the global alignment tool to calculate the similarity but it took around 10 hours to compute similarity between two genome sequences. GGDC gives best result within very small-time frame. In this GGDC web server genome sequences (FASTA format) of several bacteria are being passed as the query genome and a single genome sequence of bacteria as a reference genome is being passed as inputs. The output of the comparison is collected via e-mail. We also tried to calculate similarity between bacteria using NCBI Blast database. The protein sequences (FASTA format) of bacteria are passed as an input in the database

[7] https://www.genome.jp/tools/gn_tools_api.html

[8] https://ggdc.dsmz.de/ggdc.php#

and it gives the similarity results. But it also very tedious and the result is not accurate as compare to GGDC.

2.3 Algorithms Used

Matrix completion is a relatively matured area of research. As mentioned before, we are solving the drug-bacteria-association problem as one of matrix completion. From the known entries, some portion are assumed to be unknown and are hidden (10% around in several experimental scenarios). Matrix completion is used to recover these. Once the recovery is done, the recovered entries are compared with the ground-truth (hidden) values.

Let X is the entire drug-bacteria association matrix of size m × n denoted by $X_{m \times n}$.Here m is the number of drugs and n is the number of bacteria. The association matrix contains the binary values(1 denotes that the drug is known to act against the bacteria and 0 denotes no association or not known). The goal is to estimate X. What is available is the partially sampled version of X in Y; i.e. Y is the actual observation available. Let M be the masking operator.Then, the data fidelity term is expressed as:

$$Y = M(X) \tag{2.1}$$

This is an under-determined system, more variables (X) than the number of equations (Y). Therefore there can be infinitely many solutions. In Matrix completion the assumption is that the underlying matrix to be recovered (X) is of low rank. Under such a condition, the number of free variables are reduced and one can expect to estimate the underlying matrix Sun and Luo (2016); Candes and Plan (2010).

2.3.1 Matrix Factorization

The most elementary method to deal with the low-rank matrix completion problem is matrix factorization. It is a way to find out the latent features during the multiplication of two different entities. In this case the data matrix X is disintegrated into two latent factor matrices Um×q and Vq×n where q specifies the number of latent (hidden) factors which conclude if a drug is affiliated with a bacteria or not and the matrix X can be expressed as the multiplication of U and V. hence the equation 2.1 looks like:

$$Y = M(UV) \tag{2.2}$$

Where the data matrix X is represented by:

$$X = UV \tag{2.3}$$

And X is reconstructed by solving U and V. By minimizing the Frobenius norm of the below problem these two matrices U and V can be solved.

$$\min_{U,V} ||Y - M(UV)||_F^2 \tag{2.4}$$

this equation 2.4 is being solved by using Majorization-Minimization (MM) technique Chouzenoux *et al.* (2013); Sun *et al.* (2016). The majorization step mainly dissociate the problem (from M), so that the optimization problem can be solved by solving:

$$\min_{U,V} ||B - UV||_F^2 \tag{2.5}$$

where $B_{k+1} = X_k + \frac{1}{a} M^T(Y - M(X_k)$ and k is the number of iteration.Here, X_k is the matrix at iteration k and a is a scalar parameter used in the MM technique. The equation 2.5 has been resolved by alternating least squares method Hastie *et al.* (2015) to get U and V.In ALS (alternating least squares) algorithm while updating U, V is assumed to be constant, and U is supposed to be constant while updating V.

$$U_k \leftarrow \min_U ||B - U_{k-1}V_{k-1}||_F^2 \tag{2.6}$$

$$V_k \leftarrow \min_V ||B - U_k V_{k-1}||_F^2 \tag{2.7}$$

The complete algorithm is discussed in Mongia *et al.* (2019).

2.3.2 Nuclear Norm Minimization based matrix completion

Matrix factorization is a bilinear and leads to a non-convex minimization problem. Hence it does not guarantee global convergence. Furthermore in order to solve for matrix factorisation one needs to have an estimate on the rank of X. This is not always possible, especially for a problem such as ours. Ideally in order to estimate a low-rank matrix one needs to solve:

$$\min_X \operatorname{rank}(X) \quad \text{such that} \quad Y = M(X) \tag{2.8}$$

Unfortunately this is an NP hard problem. Theoretical investigations demonstrated that it is possible To ease the NP-hard rank penalty by its closest convex surrogate - the nuclear norm Candes and Recht (2012); Candès and Tao (2010). This leads to:

$$\min_X \|X\|_* \quad \text{such that} \quad Y = M(X) \tag{2.9}$$

Here $\|\|_*$ is the nuclear norm and is defined as the sum of the singular values of the association matrix X. Its quadratic program is solved by with the unconstrained Lagrangian version.

$$\min_X \|Y - M(X)\|_F^2 + \lambda \|X\|_* \tag{2.10}$$

Here also $\|\|_*$ denotes the nuclear norm and λ is known as the Lagrange multiplier. To solve 2.10, MM is invoked once again. Here $Q(X) = \|Y - A(X)\|_F^2 + \lambda \|X\|_*$, we can represent 2.10 in the below way in every iteration k,

$$\min_X \|B - X\|_F^2 + \lambda \|X\|_* \tag{2.11}$$

where $B_{k+1} = X_k + \frac{1}{a}M^T(Y - M(X_k)$. With the help of the inequality $||Z1 - Z2||_F >= ||s1 - s2||_2$, where s1 and s2 are singular values of the matrices Z1 and Z2 respectively, the following problem can be solved in lieu of solving the minimization problem mentioned in equation2.11:

$$\min_{s_x} \|s_B - s_X\|_2^2 + \lambda \|s_X\|_1 \tag{2.12}$$

Here, sX is the singular values of X and sB is the singular values of B and $\|s_X\|_1$ is the sum of absolute values of sX or the l1 norm.

This problem in equation 2.9 addressable via semi-definite programming, but faster algorithms like singular value thresholding Cai *et al.* (2010*b*) and singular value shrinkage Mongia *et al.* (2019) are available.

2.3.3 Deep matrix factorization

In recent days, deep learning has pervaded nearly all facets of computational science. It is an extension of matrix factorization. Instead of factoring X into two matrices, one can factor it into more. Inspired by the triumph of deep dictionary learning Tariyal *et al.* (2016),deep factorization algorithm is being recommended. In this algorithm the data matrix X is disintegrated into multiple factor matrices to acquire complicated latent structures in the data. Integrating the DMF into the equation 2.1 leads to:

$$Y = M(U1U2V) \tag{2.13}$$

Where, X is the data matrix and X is represented as:

$$X = U1U2V \tag{2.14}$$

We can consider it as the minimization problem with 3-layer matrix factorization and it is is formulated as:

$$\min_{U1,U2,V} \|Y - M(U1U2V)\|_F^2 \text{ such that } U1{>=}0, U2{>=}0 \tag{2.15}$$

This is a multi-linear problem and therefore is non-convex. There are no guarantees on global convergence. Since this is a new topic, there are no off-the-shelf algorithms for solving this. However, prior studies have extended multiplicative updates Trigeorgis *et al.* (2016) and alternating least squares Mongia *et al.* (2020) to solve it efficiently. The problem mentioned in equation 2.3 is a bilinear (Bilinearity means that it is linear in each of the variables (U and V) if the other variable kept constant (V and U respectively)) problem and equation 2.14 is a trilinear problem. They are different. Hence the features extracted from single layer matrix factorization problem are not same with the features extracted from 2-layer matrix factorization or Deep matrix factorization problem. The entire algorithm is discussed in Mongia *et al.* (2020).

2.3.4 Binary Matrix Completion

The algorithms we have discussed so far are applicable for Real matrices. However, our drug-bacteria-association problem is binary in nature. Whereas, one can apply the previous algorithms and then use some kind of thereholsing scheme to binarise the output matrix, a more natural choice is to go for 1-bit / binary matrix completion Davenport *et al.* (2014). The original paper proposed a probabilistic formulation; we are not going into the details. In the deterministic framework, we are following the problem would be posed as:

$$f(X) = \|Y - M(X)\|_F^2 + \lambda\|X\|_*, \text{s.t} X \in [0, 1] \tag{2.16}$$

where $\|\|_*$ is the nuclear norm. X is the binary matrix needs to be restored. The entire algorithm is discussed in Davenport *et al.* (2014).

2.3.5 Graph regularised matrix factorization(GRMF)

All the aforementioned algorithms are applicable to problems where the objective is to complete a partially filled matrix when no other metadata is known about the rows or columns. In our case, that is not the scenario. We have the genomic structure of bacteria and chemical structure of the drugs. From these, we have computed to similarity matrices for the rows and columns. These are easily integrated into the matrix completion framework using graph regularization. The first technique we will briefly discuss is graph regularized matrix factorization Cai *et al.* (2010*a*). In case graph regularization the model introduces a graph laplacian penalty to the cost function. This graph laplacians have been obtained from the weights between the nodes in drug/bacteria or row/column graphs and encrypt the information of row/column entities. The formulation is as follows:

$U \in \mathbb{R}^{m \times q}$ (for drugs) and $V \in \mathbb{R}^{q \times n}$ (for bacteria) which optimize the LRA objective:

$$\min_{(U,V)} \|Y - M(UV)\|_F^2 \tag{2.17}$$

Here $\|\|_F$ is the Frobenius norm and q is the number of latent features in U and V.

Sparsification of the Similarity Matrices

Sparsification of the Similarity Matrices is a procedure that is applied before graph regularization Cai *et al.* (2010a).In this case also we incorporated a p-nearest neighbour graph from each of the bacteria and drug similarity matrices, S_b, S_d. the p-nearest neighbor graph is formed as:

$$\forall i, j,\ N_{ij} = \begin{cases} 1, & \text{if } j \in N_p(i) \text{ and } i \in N_p(j) \\ 0, & \text{if } j \notin N_p(i) \text{ and } i \notin N_p(j) \\ 0.5, & \text{otherwise} \end{cases} \tag{2.18}$$

Here $N_p(i)$ represents the set of p nearest neighbors to drug d_i. N is applied to sparsify the similarity matrix S_d as:

$$\forall i, j,\ \hat{S}^d_{i,j} = N_{i,j} S^d_{i,j} \tag{2.19}$$

It produces similarity matrix (sparse matrix) for drugs. The same procedure is incorporated for the bacteria similarity matrix S_b .

Regularization is incorporated by introducing graph Laplacian penalties to the cost function of matrix factorization as shown below:

$$\min_{U,V} \|Y - M(UV)\|_F^2 + \mu_1 \operatorname{tr}(U^\top L_d U) + \mu_2 \operatorname{tr}(V L_b V^\top), \tag{2.20}$$

Where $\mu_1 > 0$ and $\mu_2 > 0$ denote the coefficients which penalize the graph regularization Laplacian terms and tr is the trace of the matrix.$L_d = D_d - S_d$ and $L_b = D_b - S_b$ are the graph Laplacians Chung (1997) for S_d(row/drug similarity matrix) and S_b(column/bacteria similarity matrix),respectively, and $D_d^{ii} = \sum_j S_d^{ij}$ and $D_b^{ii} = \sum_j S_b^{ij}$ are the associated degree matrices.The second term specifies the graph regularization for drug. The distance between latent feature vectors of two neighboring drugs is minimized by this regularization. The last(third) term is the graph regularization for bacteria. A resolution technique for the above formulation has been shown in Ezzat *et al.* (2016).

2.3.6 Graph Regularised Matrix Completion

Like matrix factorization, graph regularization can be incorporated into the nuclear norm minimization framework. In this case graph Laplacian penalties have been incorporated to consider the similarity between drugs and the similarity between bacteria. However, unlike the previous case where the factor matrices for the drugs and bacteria are well defined, we have to apply the regularization along the rows and columns of X instead. The minimization problem can be written as:

$$\min_{X} \|Y - M(X)\|_F^2 + \lambda \|X\|_* + \mu_1 \operatorname{tr}(X^\top L_d X) + \mu_2 \operatorname{tr}(X L_b X^\top). \tag{2.21}$$

where $\|\|_*$ denotes the nuclear norm. $\lambda > 0$, $\mu_1 > 0$ and $\mu_2 > 0$ denote the coefficients which penalize the graph regularization Laplacian terms and tr is the trace of the matrix. The above problem was solved with the help of ADMM (alternating direction method of multipliers) Mongia and Majumdar (2020*b*). The standard nuclear norm minimization is a convex problem and the introduced graph regularization penalties are also convex, so entire minimization problem mentioned in the equation 2.21 is convex as this is a sum of convex functions.

2.3.7 Graph Regularized Deep Matrix Factorization

Graph regularisation has also been incorporated into the deep matrix factorisation framework Sun *et al.* (2016). Since the basic deep matrix factorization model has been already discussed before, we are not repeating it. The formulation for the graph regularized three factor model is: The idea is to decompose X into more than two factor matrices. The formulation for a 3-factor model is:

$$Y = M(U1U2V) \tag{2.22}$$

U1, U2 and V are estimated by resolving the below least square problem to recover X. To deploy graph regularised Deep matrix factorization, two extra terms have been added to the standard formula using the graph Laplacian established on and bacteria entities from first and last factor matrix. The objective function is represented as:

$$\min_{U1,U2,V} \|Y - M(U1U2V)\|_F^2 + \mu_1 \operatorname{tr}(U_1^\top L_d U_1) + \mu_2 \operatorname{tr}(V L_b V^\top), \tag{2.23}$$

where $\mu_1 > 0$ and $\mu_2 > 0$ denote the coefficients which penalize the graph regularization Laplacian terms and tr is the trace of the matrix. Here it is assumed that U1 refers to the drugs and V refers to the bacteria. However, unlike the two factor model where there was a clear one-to-one mapping between drugs, bacteria and the factor matrices, the deeper model suffers from identifiability issues. One cannot say for certain if U1 pertains to drugs or if U1*U2 pertains to drugs, or if V pertains to bacteria or if U2*V pertains to bacteria. For separating the mask M from equation2.23 Majorization-Minimization Sun *et al.* (2016) is applied. Hence the new objective function is:

$$\min_{U1,U2,V} \|B - (U1U2V)\|_F^2 + \mu_1 \operatorname{tr}(U_1^\top L_d U_1) + \mu_2 \operatorname{tr}(V L_b V^\top), \tag{2.24}$$

where $B_k = (U1U2V)_{k-1} + \frac{1}{a} M^T(Y - M(U1U2V)_{k-1})$. k is the number of iterations. U1, U2 and V are solved to recover X.

$$U_1 \leftarrow \|B - (U1U2V)\|_F^2 + \mu_1 \operatorname{tr}(U_1^\top L_d U_1) \tag{2.25}$$

$$U_2 \leftarrow \|B - (U1U2V)\|_F^2 \tag{2.26}$$

$$V \leftarrow \|B - (U1U2V)\|_F^2 + \mu_2 \operatorname{tr}(V L_b V^T) \tag{2.27}$$

By iteratively executing the below update step U2 can be solved:

$$U_2 = U_1^\Gamma B V^\Gamma \tag{2.28}$$

To solve U1,equation2.25 has been differentiated w.r.t U1 and equate it to 0 which provides the below equation:

$$\mu_1 L_d U_1 + U1U2V(U2V)^T = B(U2V)^T \tag{2.29}$$

This is nothing but a sylvester equation $A_1X + XA_2 = A_3$(Here,$A_1 = \mu_1 L_d$,$A_2 =$

$U2V(U2V)^T$,$A_3 = B(U2V)^T$). The equation for V has been solved and obtained the below sylvester equation:

$$(U1U2)^T U1U2V + V(\mu_2 L_b) = (U1U2)^T B \tag{2.30}$$

The entire algorithm has been discussed in Mongia and Majumdar (2020*a*).

2.3.8 Graph Regularised Binary Matrix Completion

We mentioned before that the drug disease association prediction is a natrual application of one-bit matrix completion problem. Like other graph regularised algorithms, a prior study introduced graph regularization into the binary matrix completion framework Wishart *et al.* (2006). Similarity sparsification is also done here as the preprocessing step. The minimization problem looks like:

$$\min_X \|Y - M(X)\|_F^2 + \lambda \|X\|_* + \mu_1 \operatorname{tr}(X^\top L_b X) + \mu_2 \operatorname{tr}(X L_d X^\top), \text{s.t} X \in [0,1] \tag{2.31}$$

Where, $\lambda \|X\|_*$ denotes the nuclear norm. $\lambda > 0$, $\mu_1 > 0$, $\mu_2 > 0$. The variables have the same meaning as mentioned in other graph regularised matrix completion algorithms. To solve this equation parallel proximal algorithm (PPXA) Pustelnik *et al.* (2011) has been used. In this algorithm, X is being solved by considering a proxy variable for each of the terms in equation 2.31 and an additional proxy variable to guarantee that the scores which are being predicted are in range [0,1].

$$\hat{X}_1^{(k)} = \arg\min_X \frac{\theta}{2} \|Y - MX\|_F^2 + \frac{1}{2} \left\| X_1^{(k-1)} - X \right\|_F^2 \tag{2.32}$$

$$\hat{X}_2^{(k)} = \arg\min_X \lambda\theta \|X\|_* + \frac{1}{2} \left\| X_2^{(k-1)} - X \right\|_F^2 \tag{2.33}$$

$$\hat{X}_3^{(k)} = \min\left(\max\left(X_3^{(k-1)}, 0\right), 1\right) \tag{2.34}$$

$$\hat{X}_4^{(k)} = \arg\min_X \theta\mu_1 \operatorname{tr}(X^\top L_b X) + \frac{1}{2} \left\| X_4^{(k-1)} - X \right\|_F^2 \tag{2.35}$$

$$\hat{X}_5^{(k)} = \arg\min_X \theta\mu_2 \operatorname{tr}(XL_dX^\top) + \frac{1}{2}\left\|X_5^{(k-1)} - X\right\|_F^2 \tag{2.36}$$

Here, θ represents the number of terms considered in parallel, that is θ = 5.

- $\hat{X}_1^{(k)}$ is resolved by taking the gradient of equation 2.32 and equate to 0.

$$\theta(-M^T)(Y - M\hat{X_1}^{(k)}) + (\hat{X_1}^k - (X_1^{(k-1)}) = 0 \tag{2.37}$$

$$\theta(M^TM\hat{X_1}^{(k)}) - \theta(M^TY) + \hat{X_1}^{(k)} - X_1^{(k-1)} = 0 \tag{2.38}$$

$$(\theta M^TM + I)\hat{X}_1^{(k)} = X_1^{(k-1)} + \theta M^TY \tag{2.39}$$

Here I is the identity matrix. This equation is being resolved by the least square solutions.

- $\hat{X_2}^{(k)}$ can be calculated by the singular values(soft-thresholded) of X_2^{k-1} and multiplying the singular value matrix (thresholded) by the right and left singular vector matrices of X_2^{k-1} i.e.

$$X_2^{(k-1)} = US^{(k-1}V^T \tag{2.40}$$

$$\hat{S}^{(k-1)} = \text{soft}\left(S^{(k-1)}, \frac{\lambda\theta}{2}\right) \tag{2.41}$$

$$\hat{X}_2^{(k)} = U\hat{S}^{(k-1)}V^T \tag{2.42}$$

where $\text{soft}\left(S^{(k-1)}, \frac{\lambda\theta}{2}\right) = \text{sign}\left(S^{(k-1)}\right)\max\left(0, \left|S^{(k-1)}\right| - \frac{\lambda\theta}{2}\right)$. Here $S^{(k-1)}$ represents the singular value matrix, V and U are the right and left singular matrices of ${X_2}^{(k-1)}$, found after SVD decomposition.

- $\hat{X}_3^{(k)}$ can be solved by applying max-thresholding and after that min-thresholding on $X_3^{(k-1)}$.

- Similarly, $\hat{X}_4^{(k)}$ can be solved using the similar approach as for $\hat{X}_1^{(k)}$ and equate the gradient of 2.35:

$$2\theta\mu_1L_b\hat{X}_4^{(k)} + \hat{X}_4^{(k)} = X_4^{(k-1)} \tag{2.43}$$

$$\hat{X}_4^{(k)} = (2\theta\mu_1L_b + I)^\dagger X_4^{(k-1)} \tag{2.44}$$

- Same way $\hat{X}_5^{(k)}$ can be found as below:

$$\hat{X}_5^{(k)} = X_5^{(k-1)}(2\theta\mu_2 L_d + I)^{\dagger} \tag{2.45}$$

Where Here $A^{\dagger}$ is the Moore-Penrose pseudo inverse of A

The next iterate $X^{(k)}$ is found by taking the average of five proximal values, as below:

$$\hat{X}^{(k)} = \frac{1}{\theta}(\hat{X}_1^{(k)} + \hat{X}_2^{(k)} + \hat{X}_3^{(k)} + \hat{X}_4^{(k)} + \hat{X}_5^{(k)}) \tag{2.46}$$

With $\theta = 5$. Here each of the proxy variables is updated via the following update rule:

$$\hat{X}_i^{(k)} = X_i^{(k-1)} + 2\hat{X}^{(k)} - \hat{X}^{(k-1)} - \hat{X}_i^{(k)} \tag{2.47}$$

The complete algorithm is mentioned in Mongia *et al.* (2022).

CHAPTER 3

Results

We evaluate the proficiency of several matrix completion techniques in this section. All of the algorithms have been outlined in the Algorithms used subsection. Eight matrix completion algorithms or methods are used, which are divided into three categories given below.

- Basic frameworks (Matrix factorization(MF) Wang and Zhang (2012); Mongia *et al.* (2019) and Matrix completion(MC) or Nuclear norm minimization Mongia *et al.* (2019) and Binary Matrix completion Davenport *et al.* (2014)).
- Deep frameworks (Deep matrix factorization) Mongia *et al.* (2020)
- Graph regularized frameworks (Graph regularized matrix factorization(GRMF) Ezzat *et al.* (2016), Graph regularized matrix completion(GRMC) Mongia and Majumdar (2020*b*), Graph regularised deep matrix factorization(GRDMF) Mongia and Majumdar (2020*a*), Graph regularized binary matrix completion(GRBMC) Mongia *et al.* (2022))

Figure 3.1, illustrates the schematic flow of the proposed work. The graph regularised methods consider all three datasets(drug-bacteria association data,drug similarity and bacteria similarity) to predict the associations or to reconstruct the data matrix.

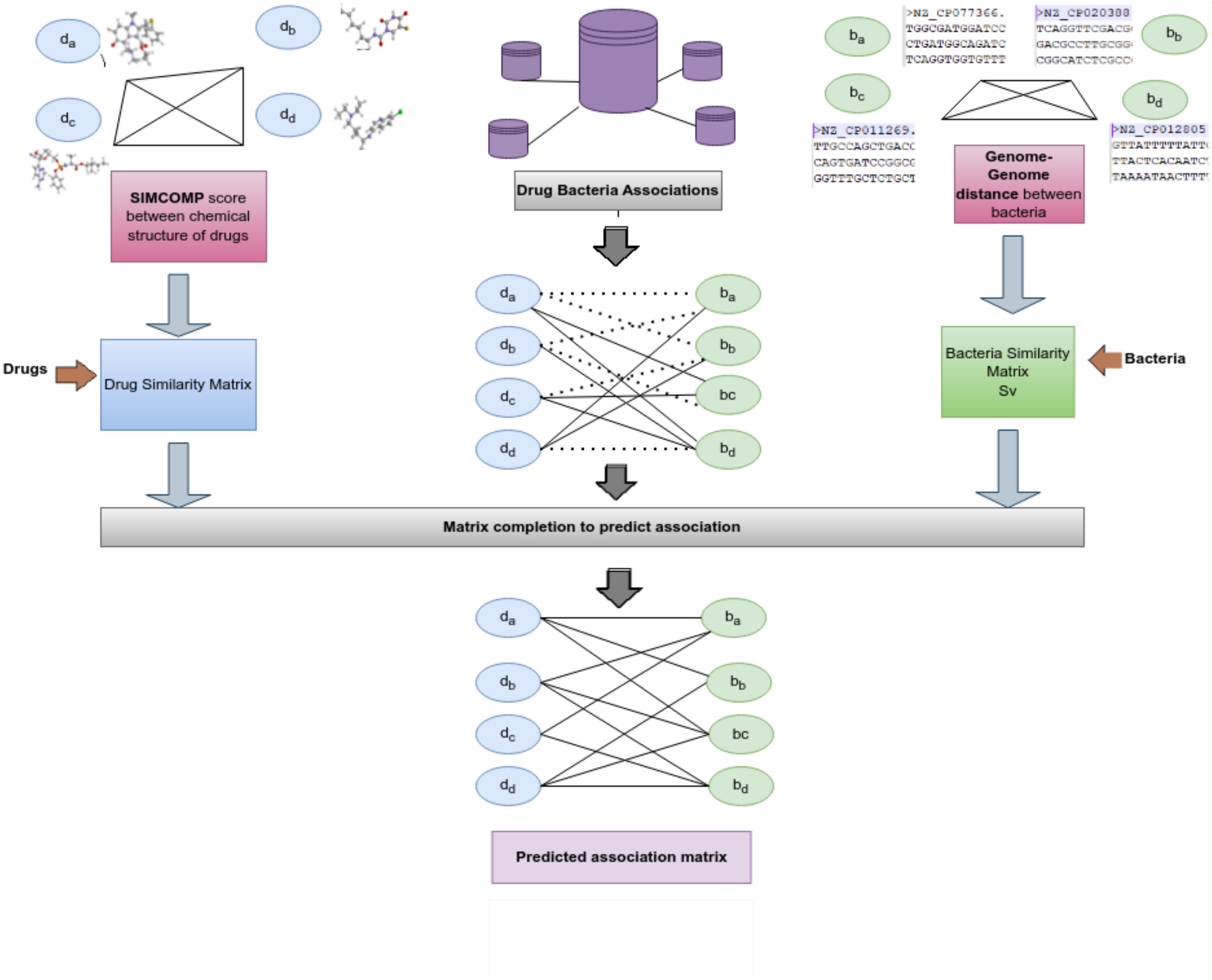

Figure 3.1: **Schematic diagram illustrating the DBA framework**

Experimental protocols

To exemplify and compare the potential of different algorithms specified above to retrieve drug-Infections associations available in our curated dataset we conduct some experimental protocols. The protocol proposes 10-fold cross validation settings(CV) on three different scenarios. We perform these 10-fold cross validation on training set and deploy eight matrix completion algorithms to predict the results.The details of experimental protocols or scenarios are mentioned below:

- 10-fold Cross validation setting on scenario1(CV1):10% of drug-bacteria association selected randomly are masked or left out as test set and the rest used as training set. This allows to determine the ability of each algorithm to predict associations between existing drugs and bacteria.
- 10-fold Cross validation setting on scenario2(CV2):10% of bacteria selected randomly are masked or left out as test set and the rest used as training set. This helps to predict drug corresponding novel bacteria,those which do not have association information.
- 10-fold Cross validation setting on scenario3(CV3): 10% of drug selected randomly are masked or left out as test set and the rest used as training set. This helps to predict bacteria corresponding novel drugs,those which have no association information.

Hyperparameter settings

The hyperparameters(step size, regularization parameters, latent factor dimensions) used in the algorithms in different scenarios are tuned using the experimental protocols described above (in the subsection Experimental protocols).We have also performed grid search on training data to select the values of the hyperparameters(step size,regularization parameter,latent factor dimensions) used in these algorithms.The hyperparameter used in CV2 are utilised to predict drug for multidrug resistance bacteria. Similarly the hyperparameters used in CV3 are utilised in predicting bacteria for any novel drug. The hyperparameters used in each scenario for different algorithms are found in tables Table 3.1,Table 3.2,Table 3.3.

Table 3.1: **Parameters used in several algorithms for CV1**

Algorithms Used	Hyperparameters for CV1
MC	delta=5.05
MF	k=5;alpha=0.5;
BMC	sigma=0.4
DMF	alpha=2; k1=25;k2=5;
GRMC	pp=7; lamda=0.03; mu1=0.02; mu2=0.005; nu1=0.4; nu2=0.3
GRMF	p=7; lambda_l = 0.05; lambda_d = 0.03; lambda_t = 0.03;
GRDMF	mu1=0.1; mu2=0.05; alpha=1; k1=25; k2=5; p=2;
GRBMC	pp=2; lamda=0.8; mu1=0.1; mu2=0.1;

Table 3.2: **Parameters used in several algorithms for CV2**

Algorithms Used	Hyperparameters for CV2
MC	delta=15.15
MF	k=20; alpha=0.05;
BMC	sigma=0.3
DMF	alpha=0.7; k1=30; k2=20;
GRMC	pp=2; lamda=0.1; mu1=0.01; mu2=1; nu1=0.4; nu2=0.1
GRMF	p=8; lambda_l = 1.5; lambda_d = 0.1;lambda_t = 0.5;
GRDMF	mu1=0.06; mu2=1.6; alpha=20; k1=25; k2=20; p=2;
GRBMC	pp=2; lamda=0.5; mu1=3; mu2=0.02;

Table 3.3: **Parameters used in several algorithms for CV3**

Algorithms Used	Hyperparameters for CV3
MC	delta=5.05
MF	k=5;alpha=0.5;
BMC	sigma=0.5
DMF	alpha=1.5; k1=20;k2=5;
GRMC	pp=2; lamda=0.01; mu1=1; mu2=0.1; nu1=0.4; nu2=0.1
GRMF	p=7; lambda_l = 0.0413; lambda_d = 0.02; lambda_t = 0.03;
GRDMF	mu1=0.09; mu2=0.01, alpha=1; k1=25; k2=5; p=2;
GRBMC	pp=2; lamda=0.8; mu1=0.08; mu2=1;

3.0.1 Experimental evaluation:

As described in the subsection Experimental protocols different experimental scenarios are adopted by hiding few data selected randomly. These masked data which is hidden or left out as test set is being predicted by the algorithms mentioned. We calculated the AUC and AUPR for all algorithms used. Table 3.4 shows the results of AUC and AUPR for all the algorithms used in CV1. Table 3.5 shows the result of AUC and AUPR for all the algorithms used in CV2. Table 3.6 shows the result of AUC and AUPR for all the algorithms used in CV3. In this subsection we illustrate and correlate the proficiency of different algorithms used to retrieve the drug-bacteria association matrix. As mentioned above in CV1 10% of association selected randomly are masked. In CV2 and CV3 10% of the entire bacteria and 10% of complete drug entities selected at random are masked. To evaluate the model two standard metrics AUC(Area under the Receiver Operating Characteristic curve) and AUPR(Area under the precision-recall curve) are considered. AUC is used to represent the ability of the model to differentiate positive and negative classes. It presumes that the classes are balanced evenly. But the problems in drug-bacteria association is highly imbalanced classes due to which AUPR is considered as more convenient metric for evaluation Ezzat *et al.* (2016); Burez and Van den Poel (2009).

Table 3.4, Table 3.5 and Table 3.6 shows the results of the experiments performed in CV1, CV2 and CV3 in terms of AUC, AUPR for all eight algorithms applied here. It is observed that graph regularised matrix factorization techniques have the better AUC, AUPR compare to non graph factorization techniques. This is expected, since the graphical versions have more information - similarity among drugs and similarity among bacteria, compared to the non graphical versions.

The scenario **CV1** depicts a problem where existing drugs can be repurposed for treating existing infections. This is not much practical importance; it has been performed since it is a standard benchmark in drug-disease-association. Here we can see that MC performs the worst; this is counter intuitive since it is mathematically a stronger technique. However, it has been observed in the past that nuclear norm minimization (MC), in practice, performs worse than matrix factorization (MF) Gu *et al.* (2010). Compared to MC, its binarized version (BMC) yields better results; this probably results from the binary nature of our problem. Both MF and DMF improve over MC and

BMC by a vast margin. However, MF performs slightly better than DMF. This may be because our dataset is small, and going deeper is resulting in overfitting. For this scenario (CV1), the graph regularized versions perform very similar to each other.

The second scenario **CV2** is the one of most practical importance. It simulates the problem where existing drugs can be treated for new bacterial infections. This is the scenario which depicts the reality of treating new strains of bacterial infections. The discussion from CV1 is pertinent here as well; we see similar trends with few slight deviations; the deeper versions perform better than the shallower counterparts.

In scenario **CV3** the problem is to check which of the existing bacterial infections are treatable with new antibiotics. This is of practical importance when new antibiotics are developed. The results show that MC performs better than both MF and BMC; we could not pinpoint the reason behind this anomaly. The graph regularized versions yield better results (as expected) than non graph regularized counterparts; the deeper versions improve over shallower ones.

Although graph regularised techniques yield enhanced outcome in this case also by delivering good AUC-AUPR value. The top performing algorithm is **GRBMC** which exhibits AUC and AUPR of $\mathbf{0.8029}$ and $\mathbf{0.5825}$ respectively.

Table 3.4: **The association prediction results by all techniques under CV1**

Metric	MC	MF	BMC	DMF	GRMC	GRMF	GRDMF	GRBMC
AUC	0.5318	0.9239	0.6890	0.9168	0.9313	0.9558	0.9277	0.9501
AUPR	0.2484	0.8270	0.4129	0.7961	0.8108	0.8928	0.8378	0.8775

Table 3.5: **The association prediction results by all techniques under CV2.**

Metric	MC	MF	BMC	DMF	GRMC	GRMF	GRDMF	GRBMC
AUC	0.4703	0.5138	0.5087	0.6049	0.7340	0.7096	0.7316	0.7301
AUPR	0.2306	0.2233	0.2099	0.3383	0.4315	0.3922	0.4236	0.4320

Table 3.6: **The association prediction results by all techniques under CV3**

Metric	MC	MF	BMC	DMF	GRMC	GRMF	GRDMF	GRBMC
AUC	0.5206	0.5018	0.4791	0.5837	0.7969	0.7904	0.7288	0.8029
AUPR	0.2521	0.3080	0.2133	0.2856	0.5523	0.5480	0.4443	0.5825

3.0.2 Prediction of drugs against unknown bacterial strain

We have taken three multi drug resistance bacteria strains(Mycobacterium tuberculosis strain NIRTX011,Klebsiella pneumoniae strain U25 and Neisseria gonorrhoeae strain H041) and predict the effective drugs against these strains with the help of graph regularised methods GRMF(graph regularised matrix factorization),GRDMF(graph regularised deep matrix factorization),GRBMC(graph regularised binary matrix completion) and GRMC(graph regularised matrix completion). The names of the predicted drugs against Mycobacterium tuberculosis strain NIRTX011 is mentioned in Table 3.7. The names of the predicted drugs against Klebsiella pneumoniae strain U25 is mentioned in Table 3.8 and the names of the predicted drugs against Neisseria gonorrhoeae strain H041 is mentioned in Table 3.9. It is observed that top predicted drugs are common in all these methods. We have also highlighted the predicted drugs which are common in all graph regularised techniques irrespective of top prediction. The predicted drugs those are common in any three graph regularised methods among four are also specified here.

Table 3.7: **Predicting drugs for Mycobacterium tuberculosis strain NIRTX011**

Algorithm Used	Recommended drugs
GRMF	Vancomycin
	Ceftriaxone
	Tazobactam
	Piperacillin
	Cefoperazone
	Sulbactam
	Teicoplanin
	Metronidazole
	Ampicillin
	Meropenem
GRDMF	Ceftriaxone
	Vancomycin
	Doxycycline
	Clavulanate
	Amoxicillin
	Tazobactam
	Piperacillin
	Levofloxacin
	Metronidazole
	Azithromycin
GRBMC	Ceftriaxone
	Vancomycin
	Amoxicillin
	Clavulanate
	Doxycycline
	Levofloxacin
	Azithromycin
	Meropenem
	Penicillin G
	Oseltamivir
GRMC	Ceftriaxone
	Vancomycin
	Doxycycline
	Clavulanate
	Amoxicillin
	Levofloxacin
	Meropenem
	Azithromycin
	Oseltamivir
	Metronidazole

Table 3.8: **Predicting drugs for Klebsiella pneumoniae strain U25**

Algorithm Used	Recommended drugs
GRMF	Tazobactam
	Piperacillin
	Vancomycin
	Amikacin
	Colistin
	Ceftriaxone
	Meropenem
	Sulbactam
	Cefoperazone
	Imipenem
GRDMF	Piperacillin
	Tazobactam
	Colistin
	Amikacin
	Ertapenem
	Levofloxacin
	Meropenem
	Sulfamethoxazole
	Trimethoprim
	Teicoplanin
GRBMC	Tazobactam
	Piperacillin
	Amikacin
	Colistin
	Meropenem
	Vancomycin
	Cefoperazone
	Sulbactam
	Imipenem
	Teicoplanin
GRMC	Piperacillin
	Tazobactam
	Colistin
	Amikacin
	Meropenem
	Sulbactam
	Cefoperazone
	Teicoplanin
	Cilastatin
	Imipenem

Table 3.9: **Predicting drugs for Neisseria gonorrhoeae strain H041**

Algorithm Used	Recommended drugs
GRMF	Ceftriaxone
	Clavulanate
	Amoxicillin
	Vancomycin
	Penicillin G
	Ampicillin
	Gentamicin
	Tazobactam
	Piperacillin
	Sulbactam
GRDMF	Ceftriaxone
	Vancomycin
	Piperacillin
	Tazobactam
	Ampicillin
	Gentamicin
	Clavulanate
	Amoxicillin
	Sulbactam
	Cefoperazone
GRBMC	Ceftriaxone
	Tazobactam
	Piperacillin
	Vancomycin
	Clavulanate
	Amoxicillin
	Sulbactam
	Cefoperazone
	Ampicillin
	Metronidazole
GRMC	Ceftriaxone
	Tazobactam
	Piperacillin
	Vancomycin
	Clavulanate
	Amoxicillin
	Ampicillin
	Sulbactam
	Cefoperazone
	Gentamicin

Table 3.10: Colour Specification used in Table B.4,B.5 and B.6

🟨	Top most predicted drug common in all methods
🟡	Drugs predicted(not top most) common in all methods
🟩	Predicted drugs common in any 3 methods
🟦	Predicted drugs common in any 2 methods

CHAPTER 4

Discussion

We have collected the bacterial information and the effective drugs and form a drug-bacteria association database (DBA). The similarity information associated with bacteria and similarity information associated with drugs are also incorporated (find in Method). On this database, several matrix completion techniques including graph regularised techniques have been deployed.

The drug-bacteria associations and the associated metadata(similarity information) are accumulated as three matrices which are drug-bacteria association matrix (Y), bacteria similarity matrix(Sb) and drug similarity matrix(Sd). There are eight matrix completion techniques are executed and analysed. The matrix completion methods which do not take into account the similarity information consider the association matrix as input (it is assumed that it is a sparse matrix from which the full low-rank association matrix will be reconstructed). It also takes into account the the masking operator where the information about the position of test and train indices are stored. The graph regularised methods not only consider the association matrix but also take into account the associated metadata in terms of similarity information. The drug similarity information is found with the help of the chemical structures of drugs and the similarity information of bacteria is found by calculating the genome distance among bacteria.These metadata along with the association matrix is passed to the graph regularised algorithms as input. It is clearly observed from Table 3.4, Table 3.5 and Table 3.6 that the graph regularised methods which considers the similarity information among bacteria and similarity information among drugs generate superior performance(AUC and AUPR is improved in case of graph regularised methods) in association prediction than the non graph regularised methods.

We have also taken few multidrug resistance bacteria strains(Mycobacterium tuberculosis strain NIRTX011,Klebsiella pneumoniae strain U25 and Neisseria gonorrhoeae strain H041) and recommend the drugs(top 10) against these strains using graph regularised method the result of which are shown in Table 3.7,Table 3.8 and Table 3.9.

CHAPTER 5

Conclusion

Computational techniques have the integral advantage as they are able to learn from the large volume of data which is important when it comes in studying drugs and bacteria. Having this ability to inspect huge amounts of data, these techniques are capable to identify potential treatments more efficiently, making it easier for clinicians to narrow down the search space for clinical trials. By this work, it can be shown that how such techniques can be utilised to predict the drugs which could act against a bacterial infection. This also helps to identify the bacteria infections treated by any newly introduced drug. This acts nothing but a recommendation system by which it is possible to recommend drug against a bacterial infection and vice versa. This work is expected to lead to new scientific ideas to re-purpose the existing drugs as antibacterial medications. It is also expected that the proposed work helps clinicians in the process of analysing and testing potential antibacterial drugs. In other words, it is believed that this work will contribute to the development of new and effective antibacterial drugs, as well as to the process of classifying and testing such drugs in a clinical setting.

REFERENCES

1. **Ashburn, T. T.** and **K. B. Thor** (2004). Drug repositioning: identifying and developing new uses for existing drugs. *Nature reviews Drug discovery*, **3**(8), 673–683.

2. **Burez, J.** and **D. Van den Poel** (2009). Handling class imbalance in customer churn prediction. *Expert Systems with Applications*, **36**(3), 4626–4636.

3. **Cai, D.**, **X. He**, **J. Han**, and **T. S. Huang** (2010*a*). Graph regularized nonnegative matrix factorization for data representation. *IEEE transactions on pattern analysis and machine intelligence*, **33**(8), 1548–1560.

4. **Cai, J.-F.**, **E. J. Candès**, and **Z. Shen** (2010*b*). A singular value thresholding algorithm for matrix completion. *SIAM Journal on optimization*, **20**(4), 1956–1982.

5. **Candes, E.** and **B. Recht** (2012). Exact matrix completion via convex optimization. *Communications of the ACM*, **55**(6), 111–119.

6. **Candes, E. J.** and **Y. Plan** (2010). Matrix completion with noise. *Proceedings of the IEEE*, **98**(6), 925–936.

7. **Candès, E. J.** and **T. Tao** (2010). The power of convex relaxation: Near-optimal matrix completion. *IEEE Transactions on Information Theory*, **56**(5), 2053–2080.

8. **Chouzenoux, E.**, **A. Jezierska**, **J.-C. Pesquet**, and **H. Talbot** (2013). A majorize-minimize subspace approach for \ell_2-\ell_0 image regularization. *SIAM Journal on Imaging Sciences*, **6**(1), 563–591.

9. **Chung, F. R.**, *Spectral graph theory*, volume 92. American Mathematical Soc., 1997.

10. **Dadgostar, P.** (2019). Antimicrobial resistance: implications and costs. *Infection and drug resistance*, 3903–3910.

11. **Davenport, M. A.**, **Y. Plan**, **E. Van Den Berg**, and **M. Wootters** (2014). 1-bit matrix completion. *Information and Inference: A Journal of the IMA*, **3**(3), 189–223.

12. **Dotolo, S.**, **A. Marabotti**, **A. Facchiano**, and **R. Tagliaferri** (2021). A review on drug repurposing applicable to covid-19. *Briefings in bioinformatics*, **22**(2), 726–741.

13. **Ezzat, A.**, **M. Wu**, **X.-L. Li**, and **C.-K. Kwoh** (2019). Computational prediction of drug–target interactions using chemogenomic approaches: an empirical survey. *Briefings in bioinformatics*, **20**(4), 1337–1357.

14. **Ezzat, A.**, **P. Zhao**, **M. Wu**, **X.-L. Li**, and **C.-K. Kwoh** (2016). Drug-target interaction prediction with graph regularized matrix factorization. *IEEE/ACM transactions on computational biology and bioinformatics*, **14**(3), 646–656.

15. **Gottlieb, A.**, **G. Y. Stein**, **E. Ruppin**, and **R. Sharan** (2011). Predict: a method for inferring novel drug indications with application to personalized medicine. *Molecular systems biology*, **7**(1), 496.

16. **Gu, Q.**, **J. Zhou**, and **C. Ding**, Collaborative filtering: Weighted nonnegative matrix factorization incorporating user and item graphs. *In Proceedings of the 2010 SIAM international conference on data mining*. SIAM, 2010.

17. **Hastie, T.**, **R. Mazumder**, **J. D. Lee**, and **R. Zadeh** (2015). Matrix completion and low-rank svd via fast alternating least squares. *The Journal of Machine Learning Research*, **16**(1), 3367–3402.

18. **Hattori, M.**, **N. Tanaka**, **M. Kanehisa**, and **S. Goto** (2010). Simcomp/subcomp: chemical structure search servers for network analyses. *Nucleic acids research*, **38**(suppl_2), W652–W656.

19. **Kanehisa, M.**, **S. Goto**, **M. Hattori**, **K. F. Aoki-Kinoshita**, **M. Itoh**, **S. Kawashima**, **T. Katayama**, **M. Araki**, and **M. Hirakawa** (2006). From genomics to chemical genomics: new developments in kegg. *Nucleic acids research*, **34**(suppl_1), D354–D357.

20. **Luo, H.**, **J. Wang**, **M. Li**, **J. Luo**, **X. Peng**, **F.-X. Wu**, and **Y. Pan** (2016). Drug repositioning based on comprehensive similarity measures and bi-random walk algorithm. *Bioinformatics*, **32**(17), 2664–2671.

21. **Meier-Kolthoff, J. P.**, **A. F. Auch**, **H.-P. Klenk**, and **M. Göker** (2013). Genome sequence-based species delimitation with confidence intervals and improved distance functions. *BMC bioinformatics*, **14**, 1–14.

22. **Meier-Kolthoff, J. P.**, **J. Sardà Carbasse**, **R. L. Peinado-Olarte**, and **M. Göker** (2022). Tygs and lpsn: a database tandem for fast and reliable genome-based classification and nomenclature of prokaryotes. *Nucleic Acids Research*, **50**(D1), D801–D807.

23. **Mongia, A.**, **E. Chouzenoux**, and **A. Majumdar** (2022). Computational prediction of drug-disease association based on graph-regularized one bit matrix completion. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, **19**(6), 3332–3339.

24. **Mongia, A.** and **A. Majumdar**, Deep matrix completion on graphs: Application in drug target interaction prediction. *In ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020*a*.

25. **Mongia, A.** and **A. Majumdar** (2020*b*). Drug-target interaction prediction using multi graph regularized nuclear norm minimization. *Plos one*, **15**(1), e0226484.

26. **Mongia, A.**, **S. K. Saha**, **E. Chouzenoux**, and **A. Majumdar** (2021). A computational approach to aid clinicians in selecting anti-viral drugs for covid-19 trials. *Scientific reports*, **11**(1), 9047.

27. **Mongia, A.**, **D. Sengupta**, and **A. Majumdar** (2019). Mcimpute: matrix completion based imputation for single cell rna-seq data. *Frontiers in genetics*, **10**, 9.

28. **Mongia, A.**, **D. Sengupta**, and **A. Majumdar** (2020). deepmc: Deep matrix completion for imputation of single-cell rna-seq data. *Journal of Computational Biology*, **27**(7), 1011–1019.

29. **Murray, C. J.**, **K. S. Ikuta**, **F. Sharara**, **L. Swetschinski**, **G. R. Aguilar**, **A. Gray**, **C. Han**, **C. Bisignano**, **P. Rao**, **E. Wool**, *et al.* (2022). Global burden of bacterial antimicrobial resistance in 2019: a systematic analysis. *The Lancet*, **399**(10325), 629–655.

30. **National Center for Biotechnology Information** (Accessed May 5, 2023). NCBI.

31. **Pustelnik, N.**, **C. Chaux**, and **J.-C. Pesquet** (2011). Parallel proximal algorithm for image restoration using hybrid regularization. *IEEE transactions on Image Processing*, **20**(9), 2450–2462.

32. **Schrader, S. M.**, **J. Vaubourgeix**, and **C. Nathan** (2020). Biology of antimicrobial resistance and approaches to combat it. *Science translational medicine*, **12**(549), eaaz6992.

33. **Shi, J.-Y.** and **S.-M. Yiu**, Srp: A concise non-parametric similarity-rank-based model for predicting drug-target interactions. *In 2015 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*. IEEE, 2015.

34. **Sun, R.** and **Z.-Q. Luo** (2016). Guaranteed matrix completion via non-convex factorization. *IEEE Transactions on Information Theory*, **62**(11), 6535–6579.

35. **Sun, Y.**, **P. Babu**, and **D. P. Palomar** (2016). Majorization-minimization algorithms in signal processing, communications, and machine learning. *IEEE Transactions on Signal Processing*, **65**(3), 794–816.

36. **Tariyal, S.**, **A. Majumdar**, **R. Singh**, and **M. Vatsa** (2016). Deep dictionary learning. *IEEE Access*, **4**, 10096–10109.

37. **Trigeorgis, G.**, **K. Bousmalis**, **S. Zafeiriou**, and **B. W. Schuller** (2016). A deep matrix factorization method for learning attribute representations. *IEEE transactions on pattern analysis and machine intelligence*, **39**(3), 417–429.

38. **Wang, Y.-X.** and **Y.-J. Zhang** (2012). Nonnegative matrix factorization: A comprehensive review. *IEEE Transactions on knowledge and data engineering*, **25**(6), 1336–1353.

39. **Wishart, D. S.**, **Y. D. Feunang**, **A. C. Guo**, **E. J. Lo**, **A. Marcu**, **J. R. Grant**, **T. Sajed**, **D. Johnson**, **C. Li**, **Z. Sayeeda**, *et al.* (2018). Drugbank 5.0: a major update to the drugbank database for 2018. *Nucleic acids research*, **46**(D1), D1074–D1082.

40. **Wishart, D. S.**, **C. Knox**, **A. C. Guo**, **S. Shrivastava**, **M. Hassanali**, **P. Stothard**, **Z. Chang**, and **J. Woolsey** (2006). Drugbank: a comprehensive resource for in silico drug discovery and exploration. *Nucleic acids research*, **34**(suppl_1), D668–D672.

41. **Zhou, Y.**, **F. Wang**, **J. Tang**, **R. Nussinov**, and **F. Cheng** (2020). Artificial intelligence in covid-19 drug repurposing. *The Lancet Digital Health*, **2**(12), e667–e676.