12-2021

Data Analytics for Google Trend Search Result of Illness Symptoms

Md Jakir Hossain

The University of Texas Rio Grande Valley

DATA ANALYTICS FOR GOOGLE TREND SEARCH RESULT OF ILLNESS SYMPTOMS

A Thesis

by

MD JAKIR HOSSAIN

Submitted in Partial Fulfillment of the

Requirements for the Degree of

MASTER OF SCIENCE IN ENGINEERING

Major Subject: Electrical Engineering

The University of Texas Rio Grande Valley

December 2021

DATA ANALYTICS FOR GOOGLE TREND SEARCH RESULT OF ILLNESS SYMPTOMS

A Thesis
by
MD JAKIR HOSSAIN

COMMITTEE MEMBERS

Dr. Weidong Kuang
Chair of Committee

Dr. Wenjie Dong
Committee Member

Dr. Young Zhou
Committee Member

December 2021

ABSTRACT

Hossain, Md Jakir, DATA ANALYTICS FOR GOOGLE TREND SEARCH RESULT OF ILLNESS SYMPTOMS. Master of Science in Engineering (MSE), December 2021, 47 pp., 5 table, 12 figures, 32 references.

The outbreak of COVID-19 has escalated from March 2020. Since then, people from all over the world are curiously searching different types of illness symptoms including corona virus. Before the outbreak, people were also searching different types of symptoms at different times. Both diseases share same symptoms, but in the flu season. If certain types of symptoms are visible at summer season, then these symptoms are for corona virus. The main purpose of our study is to find out this discriminative information from these search result. We will discuss some mathematical concepts and then develop an algorithm based on those formulas and then apply this algorithm to those datasets to find out those discriminative data.

TABLE OF CONTENTS

LISTS OF TABLES

LIST OF FIGURES

CHAPTER I

INTRODUCTION

Our study is about finding out discriminative symptoms between COVID-19 and seasonal flu. Though both disease groups have lots of common symptoms such as fever, cough, and fatigue, but only a few symptoms are more prominent in COVID-19; e.g., anosmia. These symptoms are also visible for seasonal flu at a certain duration. We will consider those conditions to design our mathematical model for sorting out those symptoms. The target dataset contains higher number of search results of those information at summer season and the background dataset contains lower number of search results at that time. The variance of those symptoms will be higher if we divide the target search results with the background search result.

We use Google Trends search results as our background and target datasets. The epidemic of COVID-19 began after March 2020. We will consider search results of 2019 as background dataset when corona virus was not available. The search results of 2020 as target dataset. Both datasets contain 400 columns of different symptoms and 365 rows for each day search result of each states of USA. As we are going to find out important symptoms comparing those two datasets, so we have to at first mathematically design the problem and then build up our own algorithm to get the result.

CHAPTER II

REVIEW OF LITERATURE

2.1 Non Negative Matrix Factorization (NMF)

Non Negative Matrix Factorization is very useful for dimensionality reduction and feature selection. As we are going to sort out most important symptoms from a specified set of features. We will use this methodology in our study.

For our study, we are considering two datasets. The first dataset X is target dataset and the second dataset Y is background dataset. The first step of non negative matrix factorization is making both of those datasets as zero mean datasets. Then we need to make them independently square matrices where we will have covariance of the original matrices and right eigen vectors. After then we will need to multiply the square matrix of the target dataset with the inverse of the square matrix of the background dataset. This multiplication will increase the variance of the target dataset. Though we are using covariance matrices of target and background datasets, but it will keep original information of both datasets. Since the new matrix contains the variance matrix and the right eigen vectors of those datasets. We can decompose this new matrix into two parts by applying eigen value decomposition(evd) and singular value decomposition(svd) methods. As we need to make two non negative matrices, so at first, we have to apply eigen value decomposition method. Though svd is almost like to the evd, the difference is eigen vector. The eigen vector of evd contains only positive elements but for svd it contains both positive and negative elements. It will be easier to inspect and sort out important features, if we can convert the single matrix into two non negative matrices.

Let matrix Z be the product of the matrices W and H,

$$Z = W*H$$

Here Z is the product matrix of the multiplication of diagonal matrix of eigen values with eigen matrix and the inverse of eigen matrix. The inverse and transpose of eigen matrix will be similar because the matrix is orthogonal. We will discuss this decomposition process in the next section. Matrix multiplication can be implemented as computing the column vectors of Z as linear combinations of the column vectors in W using coefficients supplied by columns of H. That is, each column of Z can be computed as follows:

$$z_i = W*h_i$$

Where z_i is the *i*-th column vector of the product matrix Z and h_i is the *i*-th column vector of the matrix H. When multiplying matrices, the dimensions of the factor matrices may be significantly lower than those of the product matrix and it is this property that forms the basis of NMF. NMF generates factors with significantly reduced dimensions compared to the original matrix.

The mat lab command for NMF is –

[W,H] = nnmf(Z,k)

This command factorizes *n* by *m* matrix Z into two nonnegative factors W (*n* by k) and H (k by *m*). The factorization creates lower rank than that of the original matrix. Consequently, there will exist some error. We can minimize the error between Z and W*H by applying alternating least square method.

2.2 Eigen Value Decomposition

This decomposition technique is the basic building block of our experiment. If we can understand this mathematical explanation, it will be easier for us to understand other two decomposition techniques. Other two techniques also mostly follow the same concepts of eigen value decomposition. In our experiment, firstly we will make zero mean datasets. This will help us compare two datasets in the same ground. We have used matlab built in function to do so. In below, Xn and Yn are zero mean target and background datasets where μ_x and μ_y are mean values of X and Y.

$$Xn = X - \mu_x$$

$$Yn = Y - \mu_y$$

After then we will make square matrices of those two zero mean datasets. The dot product of any matrix with its transpose matrix will provide us this output. At this point, we will divide the zero mean square matrix of target dataset with that of background dataset. This division will increase the variances of symptoms of corona virus when it is not normal flu season.

$$\text{Xcov} = \text{Xn}^{T} * \text{Xn}$$

$$\text{Ycov} = \text{Yn}^{T} * \text{Yn}$$

$$\text{Z} = (\text{Ycov})^{-1} * \text{Xcov}$$

Here, we will briefly discuss how mathematically we can get eigen values and their corresponding vectors. We know the characteristic equation of Z is - $\Delta(\lambda) = \det(Z - \lambda I)$. The roots of $\det(Z - \lambda I) = 0$ are the eigenvalues of Z. A real or complex number λ is called an eigenvalue of the $n \times n$ real matrix Z if there exists a nonzero vector U such that $Z U = \lambda U$. Any vector U satisfying $Z U = \lambda U$ is called a (right) eigenvector of Z associated with eigenvalue λ.

$$Z * U = \lambda * U$$

$$Z = U * \lambda * U^{T}$$

$$Z = W * H$$

$$[U, \lambda] = \text{eig}(Z)$$

2.3 Orthonormalization

The Euclidean distance of a vector from the origin is a norm, called the Euclidean norm, or 2-norm, which may also be defined as the square root of the inner product of a vector with itself.

$\|U\|_2 = \sqrt{UU^T} = \sqrt{\sum_{i=1}^{n} ||U_i||^2}$

A vector U is said to be normalized if its Euclidean norm is 1 or $U^T U = 1$.

Two vectors U_1 and U_2 are said to be orthogonal if $U_i^T U_j = 0$ for $i \neq j$.

Two vectors U_1 and U_2 are said to be orthonormal if $U_i^T U_j = 0$ for $i \neq j$ and $U_i^T U_j = 1$ for $i = j$.

Given a set of linearly independent vectors $\{e_i , 1 \leq i \leq m\}$, an orthonormal set $\{q_i , 1 \leq i \leq m\}$ can be obtained by the Schmidt orthonormalization procedure:

$u_1 = e_1;\ q_1 = \frac{u_1}{||u_1||}$

$u_2 = e_2 - {q_1}^T e_2\ q_1;\ q_2 = \frac{u_2}{||u_2||}$

$u_m = e_m - \sum_{k=1}^{m-1} {q_k}^T\ e_m\ q_k;\ q_m = \frac{u_m}{||u_m||}$

$q_k = \frac{u_k}{||u_k||}$

2.4 Alternating Least Square

The singular value decomposition of Z is -

$Z = U\Sigma V^T + N$

$Z = L + N$

$L = U\Sigma V^T$

$= WH$

Where, $W = U\Sigma$ and $H = V^T$

The loss function for matrix factorization –

$$\underset{W,H}{argmin} ||Z - WH||_F^2$$

If we want to minimize error between input and output then we have to consider the first derivative of this loss function as zero.

First we have to consider H as constant and differentiate the term with respect to W and update the value of W.

$$\frac{d}{dW} ||Z - WH||_F^2$$

$\frac{d}{dW} ||Z - WH||_F^2 = 0$

or $2(X - WH)(-H) = 0$

or $WH^2 - XH = 0$

or $WH^2 = XH$

or $W = XH(H^{-2})$

or W = X(H^{-1})

or W = X$H^T(HH^T)^{-1}$

In a similar way

H = X$W^T(WW^T)^{-1}$

Using alternating least square methods we can update W and H to minimize error between input and output.

2.5 Singular Value Decomposition (SVD)

SVD is the most important decomposition technique in our study. This method is a little bit different from the eigen value decomposition. In eigen value decomposition, we only have diagonal matrix of eigen values and the right eigen vector. In SVD, we will have one diagonal matrix of singular values and two singular vectors. The first one is right singular matrix and the second one is left singular matrix. Both matrices are orthonormal. The determinants of both matrices are one. The inverse and transpose of those matrices same. We are going to derive mathematical explanation of this decomposition in below.

We already have got ideas how to get eigen values and eigen vector from the characteristic equation of our mixed matrix Z. from eigen value decomposition, we can write –

$$Z*U = U*\lambda$$

$$\text{Or } Z = U * \lambda * U^T$$

We will apply right and left dot product in our mixed matrix Z. It will give us right singular value matrix, left singular value matrix and the diagonal singular value matrix.

The right dot product is -

$$Z^T*Z = U_1 * \lambda^T * U_1^T*U_1* \lambda*U_1^T$$

$$= U_1 * \lambda^2 * U_1^T$$

Since U_1 is orthonormal matrix and λ is diagonal matrix, $U_1^T*U_1$ = 1 and $\lambda^T* \lambda = \lambda^2$.

Similarly, the left dot product is -

$$Z*Z^T = U_2* \lambda*U_2^T * U_2 * \lambda^T * U_2^T$$

$$= U_2 * \lambda^2 * U_2^T$$

If we consider $U_1 = U_2$ then we can write $Z*U_2 = U_1 * \lambda^2$ in a similar way of Z*U = U*λ.

$$Z = U_1 * \lambda^2 * U_2^T$$

Here λ^2 is the diagonal matrix of singular values and U_2^T is the right singular vector matrix. The columns of right singular vector matrix are corresponding vectors for singular values. We can order our specified features using values of first singular vector. Since U_1 and U_2 are not same matrix, there will exist some errors in this process. The alternating least square method is used to minimize error by updating those matrices.

The mat lab command for singular value decomposition is – [U1,S, U2] = svd(Z);

2.6 Principal component analysis (PCA)

The purpose of Principal component analysis is to order its features according to the greatest variance. The column of highest variance data will appear on the first coordinate, the second greatest variance data on the second coordinate, and so on. All of those coordinates are orthogonal to each other.

Our main goal of this project is to order 11 specific features according to the highest variance. This procedure is the final mathematical step in our project. We already have zero mean target dataset, diagonal matrix of singular values and right eigen vector of our mixed matrix. At this point we have used built in mat lab function to find out the index of our greatest singular value. Again, we have used another mat lab command to find out the eigen vector of this principal component. Finally, we have multiplied the zero mean target dataset with this eigen vector. This will provide us the order of our 11 symptoms according to the highest variance.

The order of features can be obtained by following equation –

```
[~, ind] = sort(diag(S), 'descend');
```

This built-in mat lab function will sort out columns of singular values according to the descending order. The column of highest singular value will appear on the first place and the second one will appear on next and so on. It will give us our desired order of features. We also need to understand the strength of our singular values. The right singular vector of this highest singular value will give us strength of all of these singular values or features of our zero mean target dataset. We can obtain this graphical representation by using the following mat lab command –

Up = U2(:, ind(1:d));

This last step is multiplication of zero mean target dataset with the principal eigen vector. This multiplication will help us update value of our target dataset according to the strength of principal component.

Xp = Xn*Up

The below defined function is our algorithm which we have developed to search for discriminative information from google trend search result. All of those methods we have discussed earlier are used in this defined function.

function [Up, Xp] = dpca(X,Y,d)

2.7 Tensor Decomposition

2.7.1 Tensor

Tensors (of order higher than two) are arrays indexed by three or more indices, say (i, j, k, · · ·) – a generalization of matrices, which are indexed by two indices, say (r, c) for (row, column). Matrices are two-way arrays, and there are three- and higher-way arrays (or higher-order) tensors.

In case of tensor decomposition, singular value decomposition plays a vital role. Singular value decomposition is two dimensional but tensor decomposition is three dimensional.

2.7.2 Some useful products

Two matrix products are useful for tensor decomposition. The first one is Kronecker product and another one is khatri Rao product.

1) Kronecker product: if we want to multiply two matrices then the column to column and row to row product will happen. That means, if A is I by J and B is K by L matrices then their product C will be IK by JL matrix – every element of first matrix will multiply every column of second matrix.

   A ⊗ B = [BA(1,1) … BA(1,K); BA(2,1) … BA(2,K); … … … ; BA(I,1) … BA(I,K)]

   The most useful property of Kronecker product is $b^T \otimes \mathrm{a} = \mathrm{a}b^T$ and the useful property of vec() function is – it makes an I by J matrix into a IJ by 1 vector. If we consider three matrices of $\mathrm{svd(Z)} = \mathrm{VD}U^T$ and then apply both properties of Kronecker product and vec() function, we will get –

   $$\mathrm{Vec}(\mathrm{VD}U^T) = (\mathrm{U} \otimes \mathrm{V})\ \mathrm{vec(D)}.$$

   The cost function or Euclidean distance for Kronecker product is –

   $$\min_D ||Z - \mathrm{VD}U^T||_F^2 = \min_d ||vec(Z) - (\mathrm{U} \otimes \mathrm{V})\mathrm{d}||_F^2$$

2) Khatri Rao product: The Khatri Rao product is the column wise multiplication of two equal size matrices. If we want to multiply a matrix U of I by J with a matrix V of K

by J, then the Khatri Rao product will be IK by J. This will reduce the number of elements. As a result the performance will be higher.

Let matrix U = $[u_1, \ldots, u_j]$ and V = $[v_1, \ldots, v_j]$ then the Khatri Rao product of U and V is –

$$U \circledcirc V = [u_1 \otimes v_1, \ldots, u_j \otimes v_j]$$

If we consider three matrices of svd(Z) = VDU^T and then apply properties of vector function and the Khatri Rao product, we will get –

$$Vec(VDU^T) = (U \circledcirc V)\ vec(D).$$

The cost function or Euclidean distance for Khatri Rao product is –

$$\min_D ||Z - VDU^T||_F^2 = \min_d ||vec(Z) - (U \circledcirc V)d||_F^2$$

2.7.3 Rank Decomposition for Tensors

The property of outer product is that when we multiply two vectors together it will produce a rank one matrix. The rank of a matrix determines the linearly independent columns or rows in it.

A rank-1 third-order tensor Z of size I by J by K is an outer product of three vectors: Z(i, j, k) = a(i)b(j)c(k), ∀i ∈ {1, · · · , 365 }, j ∈ {1, · · · , 400 }, and k ∈ {1, · · · , 51 }; i.e., Z = a◉b◉c. In our study, we will not go beyond the rank 1 three order tensor. But for proper understanding how tensor works and how cost function converges we will discuss rank three tensor here. The rank of tensor Z is the minimum number of rank-1 tensors needed to produce Z as their sum. Therefore, a tensor of rank at most F can be written as –

$$Z(i, j, k) = \sum_{f=1}^{F} a_f(i) \circledcirc b_f(j) \circledcirc c_f(k)$$

If we consider Z as rank 3 and k = 1, then we will get

$$Z(i, j, 1) = \sum_{f=1}^{F} a_f(i) \circledcirc b_f(j) \circledcirc c_f(1)$$

$$= a_1(i) \circledcirc b_1(j) \circledcirc c_1(1) + a_2(i) \circledcirc b_2(j) \circledcirc c_2(1) + a_3(i) \circledcirc b_3(j) \circledcirc c_3(1)$$

As we are considering $c_f(1)$ as a constant value and we know $a_f \circledcirc b_f = b_f {a_f}^T$

$$Z(:, :, 1) = \sum_{f=1}^{F} b_f {a_f}^T c_f(1)$$

$$= b_f {a_f}^T c_f(1)$$

$$Z(:, :, 2) = b_f {a_f}^T c_f(2)$$

$$\ldots$$

$$Z(:, :, k) = b_f {a_f}^T c_f(k)$$

$$= b_f {a_f}^T diag(c_f(k))$$

We can write $c_f(k)$ as $diag(c_f(k))$ because it is a constant matrix.

If we apply vectorization property of Khatri Rao product on all of those independent vectors then we will get them together in the below way –

$$\text{Vec(Z(:, :, k))} = \text{vec}(b_f {a_f}^T diag(c_f(k)))$$

$$[\text{Vec(Z(:, :, 1)), Vec(Z(:, :, 2)), …, Vec(Z(:, :, 51))}] = a_f \circledcirc b_f \text{vec}(diag(c_f(k)))$$

$$= a_f \circledcirc b_f c_f$$

$$= a_f \circledcirc b_f {c_f}^T$$

We can write c_f as ${c_f}^T$ because it is a diagonal constant matrix.

If we apply again the vectorization property, then we will get third order Z_3 (as we have considered the third face as constant) –

$$Z_3 = a_f \circledcirc b_f \circledcirc {c_f}^T$$

In a similar way, we will get Z_2 and Z_1, if we consider 2nd face and 1st face as constant.

$$Z_2 = a_f \circledcirc c_f {b_f}^T$$

$$Z_1 = b_f \circledcirc c_f {a_f}^T$$

The cost function for first, second and third face of the tensor is –

$$\min_{a_f} ||Z_1 - b_f \circledcirc c_f {a_f}^T||_F^2$$

$$\min_{b_f} ||Z_2 - a_f \circledcirc c_f {b_f}^T||_F^2$$

$$\min_{a_f} ||Z_3 - a_f \circledcirc b_f {c_f}^T||_F^2$$

CHAPTER III

METHODOLOGY AND FINDINGS

3.1 Datasets

In our study, we have used open-source data. These data sets can be freely used for research purpose. We have downloaded our datasets from Google Trends search result. The first one is the background dataset which is search result of 2019 or 2018. This dataset contains less search result about COVID-19 symptoms because the epidemic did not happen at that time. People did not search that much of these three corona virus symptoms online at that time. The second one is the target dataset which is search result of 2020 and it contains more information of COVID-19 symptoms because people are more aware about those three symptoms after the outbreak of pandemic. Though both datasets contain almost same frequency of seasonal flu search results, but the frequency of search result of corona virus three symptoms is higher in the dataset of 2020 search result than that of the background dataset of 2019 or 2018. Both datasets have 400 columns and 365 rows for each state of USA. All 51 states of USA contain separately 400 different symptoms and 365 days frequency of search results.

Though, many symptoms are common between COVID-19 and seasonal flu, but we will mainly focus on 11 important symptoms. Of which, 8 symptoms are similar between them and 3 symptoms are only for COVID-19. These 3 symptoms are stronger in 2020 dataset and not much stronger in 2019 dataset. Again, both datasets contain different frequencies for different symptoms search result over the 12-month period. We know June to October is summer season. In this season, the search result of COVID-19 symptoms will be higher. Normal flu season is considered between November and May. Both diseases share same symptoms in this flu season. Both datasets will have almost same frequency of search result of those 11 symptoms in this flu season. It will be very difficult to differentiate these three symptoms are for COVID-19 or for seasonal flu as both diseases share same symptoms in this season.

If we can properly understand our datasets earlier, it will be easier for us to apply our algorithm efficiently. We can make good analysis of our experimental output.

In our experiment, we will consider our background dataset as a matrix $\{y_i \in R^D\}_{i=1}^{n}$, which contains the information of flu, e.g., Google Trends data in 2019 or 2018 when there was no COVID-19 but flu, and target dataset as a matrix $\{x_i \in R^D\}_{i=1}^{m}$, which contains the information of both flu and COVID-19, e.g., Google Trends data in 2020. Here, D denotes the number of 11 searched symptoms and i is time index for one month or for one year. Both datasets will be used as matrices and contain the same number of rows and same number of columns. We will consider every column as a vector. Then we will compare these 11 target vectors with respect to the same vectors of the background dataset.

3.2 Discriminative Algorithm

We have developed our discriminative algorithm by using several built-in functions. We have explained working principles of those built-in functions mathematically in our literature review section. We have briefly summarized our discriminative algorithm in the following table -

Table 1: Discriminative Algorithm

Discriminative Algorithm
1. Input: Zero-mean target and background data $\{x_i\}_{i=1}^{m}$ and $\{y_i\}_{i=1}^{n}$; number of dimensions $d = 11$. 2. Construct covariance matrices of $\{x_i\}$ and $\{y_i\}$ to obtain C_x and C_y. 3. Perform matrix decomposition on ${C_y}^{-1}C_x$ to obtain the two factorization components W and H. 4. Output: W and H.

The first step of our algorithm is to make zero mean target and background datasets. It will help us compare matrices in the same ground. The second step is to make square matrices that will help us to obtain covariance matrices of zero mean target and background datasets. The next step is to divide covariance matrix of zero mean target dataset with that of background dataset. We will then decompose this mixed matrix into two matrices W and H. Every element of each column of H matrix contain information for each 11 symptoms. We will only consider the single column of it for the highest singular value of diagonal matrix of all 11 singular values.

3.3 Results for target data 2020 and background data 2019

We know the target dataset contains higher information for corona virus symptoms. The searching frequency for corona virus symptoms is less in background dataset. Before applying our algorithm, we know earlier that the variation of COVID-19 symptoms will be higher than other eight symptoms. We will apply our algorithm for overall USA search results and for a single state. Both cases, we will expect the same result. In this case we have considered 2020 search results as target dataset and 2019 search results as background dataset. We know the summer season is between June and March, the normal flu season is other months. We will observe variances of those symptoms in summer season and in normal flu season.

Before March 2020 people were not searching COVID-19 symptoms at summer season. People were searching more about those symptoms at summer season after the outbreak of corona virus after March 2020. As people were affected with the corona virus, people did search those symptoms to understand about their health condition. If we apply our algorithm for overall USA data and individually on every state, the variance of COVID-19 symptoms will be higher at summer season. The serial of those 3 symptoms will appear in first second and third position of 11 position for total 11 symptoms. In flu season, both diseases show these 11 symptoms. Before March 2020 and after March 2020, both flu seasons contain similar information of those symptoms. Variations of those symptoms will not maintain the order at normal flu season.

3.3.1 Symptoms strength for overall USA in June

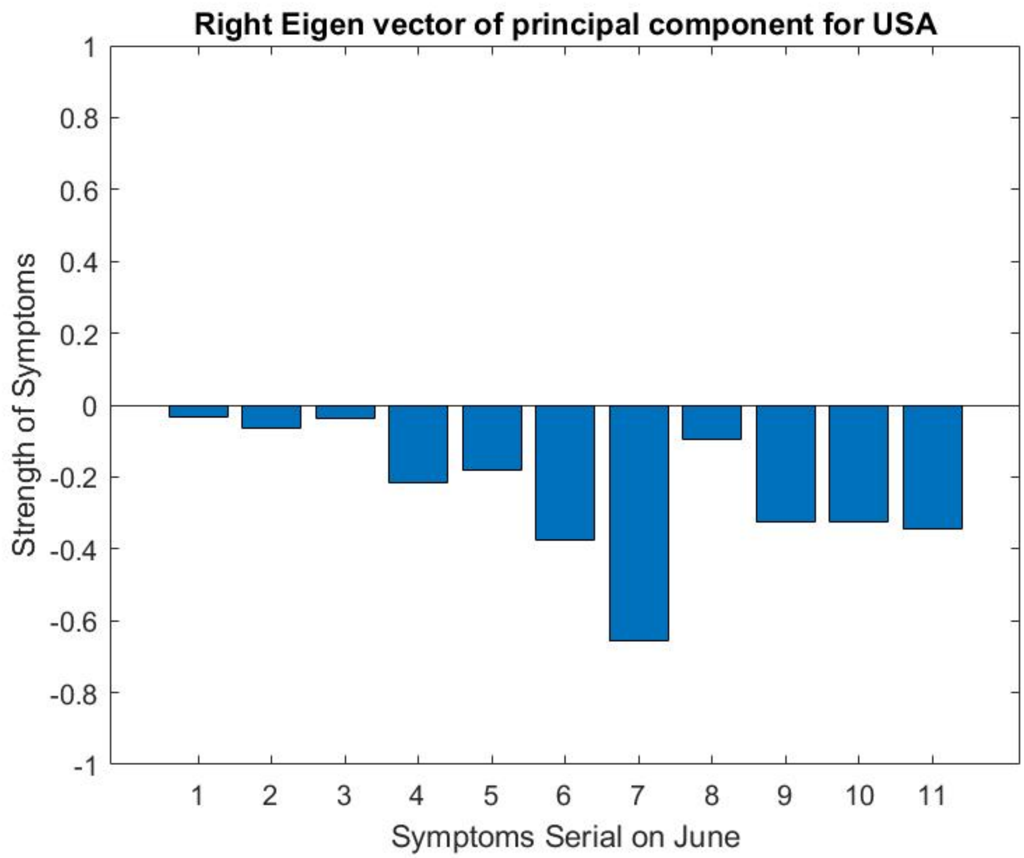

Figure 1: Coefficients for USA in June 2019

This dataset represents single vector column of our right singular vector matrix for 11 symptoms of COVID-19 and seasonal flu. All bar graph values are coefficients of 11 symptoms according to the principle singular value of our diagonal matrix of singular values. First three values are for COVID-19 symptoms of Ageusia, Shortness of Breath, and Anosmia. Values of coefficients of those three symptoms are higher than that of the rest other symptoms at summer season for overall USA.

3.3.2 Symptoms strength for overall USA in March

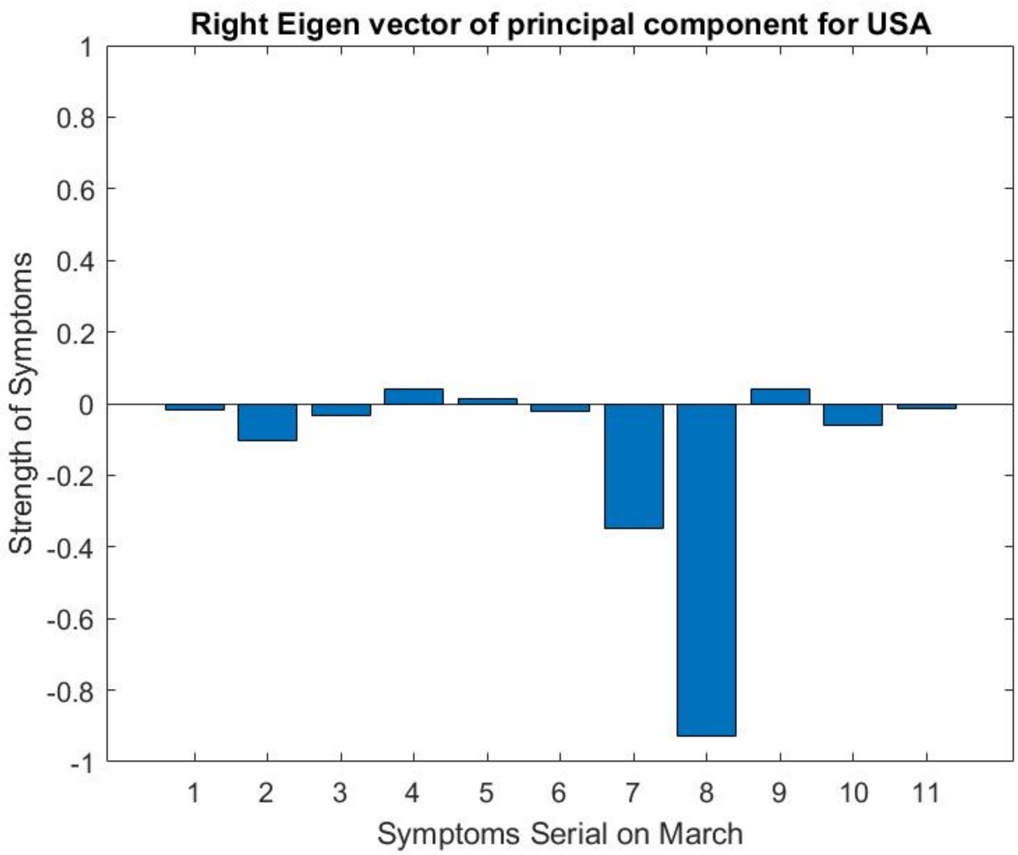

Figure 2: Coefficients for USA in March 2019

This dataset represents single vector column of our right singular vector matrix for 11 symptoms of COVID-19 and seasonal flu. All bar graph values are coefficients of 11 symptoms according to the principle singular value of our diagonal matrix of singular values. First three values are for COVID-19 symptoms of Ageusia, Shortness of Breath, and Anosmia. We can see values of coefficients of those three symptoms are not only higher values at normal flu season for overall USA.

3.3.3 Symptoms strength for Florida in June

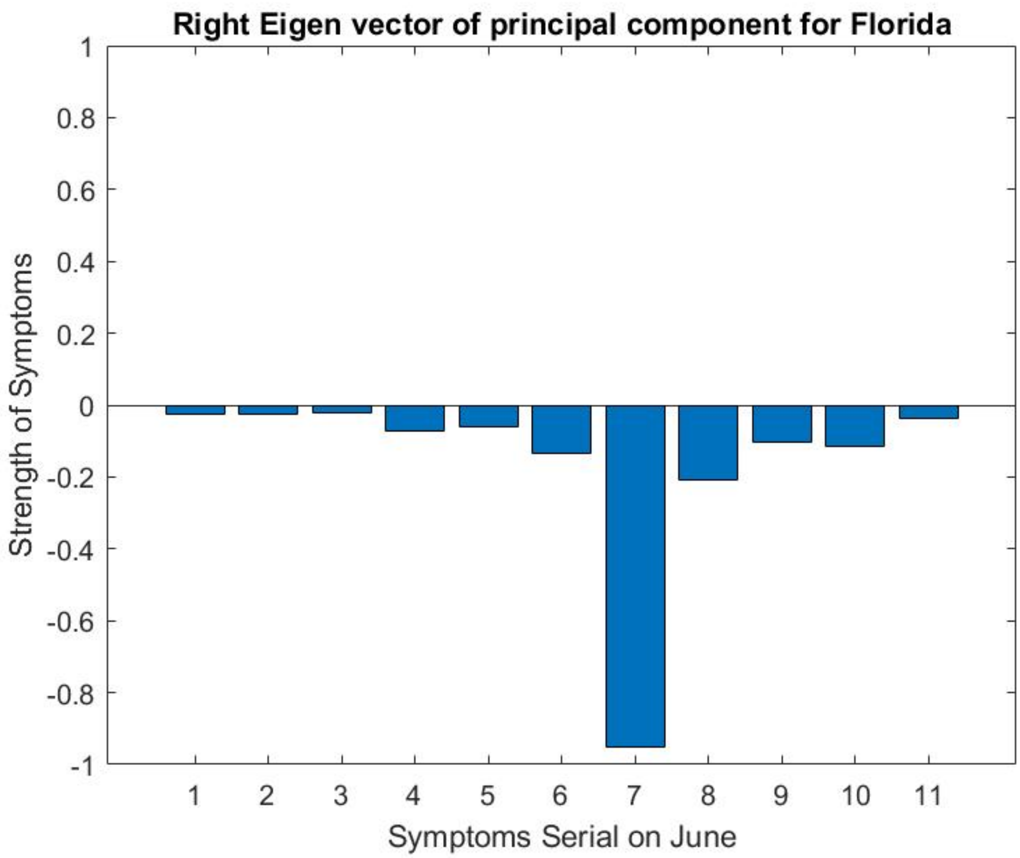

Figure 3: Coefficients for Florida in June 2019

This dataset represents single vector column of our right singular vector matrix for 11 symptoms of COVID-19 and seasonal flu. All bar graph values are coefficients of 11 symptoms according to the principle singular value of our diagonal matrix of singular values. First three values are for COVID-19 symptoms of Ageusia, Shortness of Breath, and Anosmia. Values of coefficients of those three symptoms are higher than that of the rest other symptoms at summer season for the single state Florida.

3.3.4 Symptoms strength for Florida in March

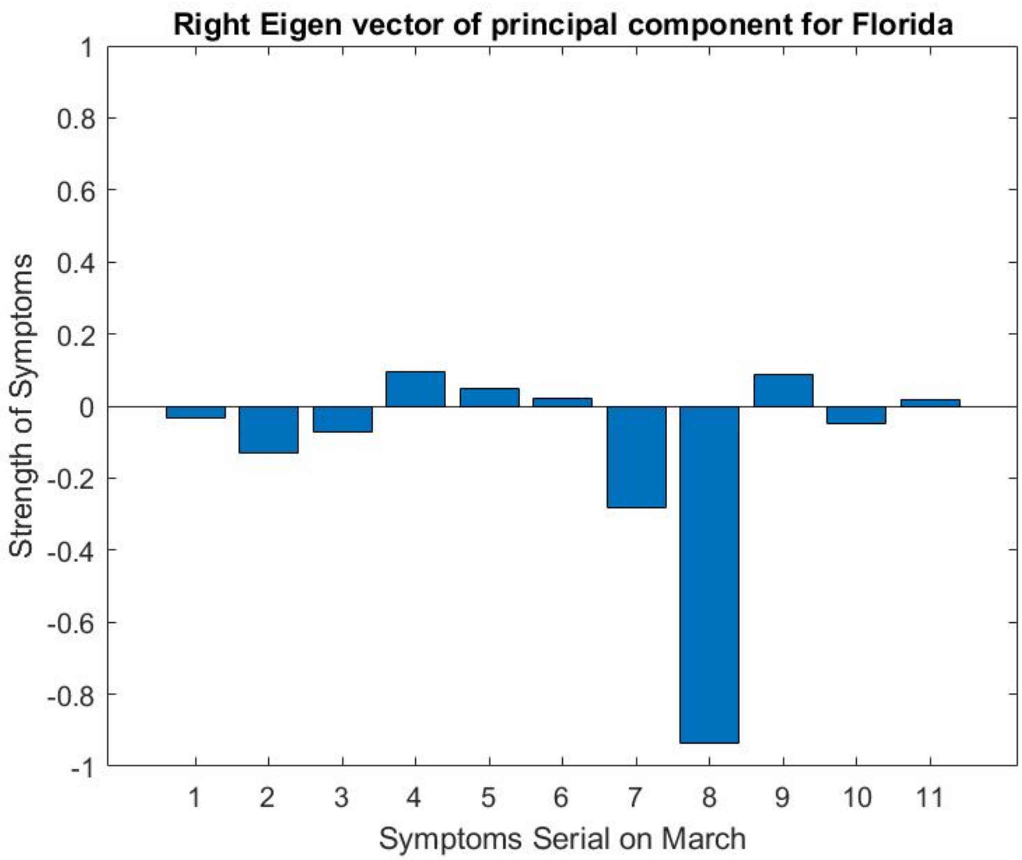

Figure 4: Coefficients for Florida in March 2019

This dataset represents single vector column of our right singular vector matrix for 11 symptoms of COVID-19 and seasonal flu. All bar graph values are coefficients of 11 symptoms according to the principle singular value of our diagonal matrix of singular values. First three values are for COVID-19 symptoms of Ageusia, Shortness of Breath, and Anosmia. We can see values of coefficients of those three symptoms are not only higher values at normal flu season for the single state Florida.

3.3.5 Symptoms position for 6 months for USA

Table 2: Coefficients Position for USA in 2019

	Serial of Symptoms
June	1 2 3 8 5 4 10 6 9 11 7
July	2 1 3 5 10 4 11 9 6 7 8
August	2 3 1 8 4 5 9 11 10 6 7
December	11 4 9 10 7 6 2 5 8 1 3
January	7 8 4 6 9 11 10 5 2 1 3
February	7 8 6 4 3 2 10 1 9 11 5

We have applied our algorithm for overall USA dataset for 6 months period. The above list represents position of our 11 symptoms when background dataset is 2019 search result, and the target dataset is 2020 search result. All positions in the above list are ordered in highest to lowest from left to right. Numbers are for columns in which we have assigned our 11 symptoms. In list, 1, 2 and 3 stands for symptoms Ageusia, Shortness of Breath, and Anosmia. June, July and August, these three months are summer season and December, January, February are normal flu season. Since the variance of COVID-19 symptoms are higher in summer season, these three symptoms have appeared in the 1st, 2nd, and 3rd position in the summer season. In normal flu season, the appearance of those symptoms does not maintain the order because their variances are not comparatively that much higher than that of other symptoms.

3.3.6 Symptoms position for 6 months for Florida

Table 3: Coefficients Position for Florida in 2019

	Serial of Symptoms
June	3 1 2 11 5 4 9 10 6 8 7
July	1 3 2 5 4 6 11 9 10 7 8
August	1 3 2 5 4 10 7 6 9 11 8
December	11 9 2 6 4 1 5 3 10 8 7
January	10 11 8 9 4 2 3 7 1 6 5
February	7 8 11 4 6 9 2 10 3 1 5

We have applied our algorithm for a particular state Florida dataset for 6 months period. The above list represents position of our 11 symptoms when background dataset is 2019 search result, and the target dataset is 2020 search result. All positions in the above list are ordered in highest to lowest from left to right. Numbers are for columns in which we have assigned our 11 symptoms. In list, 1, 2 and 3 stands for symptoms Ageusia, Shortness of Breath, and Anosmia. June, July and August, these three months are summer season and December, January, February are normal flu season. Since the variance of COVID-19 symptoms are higher in summer season, these three symptoms have appeared in the 1^{st}, 2^{nd}, and 3^{rd} position in the summer season. In normal flu season, the appearance of those symptoms does not maintain the order because their variances are not comparatively that much higher than that of other symptoms.

3.4 Results for target data 2020 and background data 2018

We know the target dataset contains higher information for corona virus symptoms. The searching frequency for corona virus symptoms is less in background dataset. Before applying our algorithm, we know earlier that the variation of COVID-19 symptoms will be higher than other eight symptoms. We will apply our algorithm for overall USA search results and for a single state. Both cases, we will expect the same result. In this case we have considered 2020 search results as target dataset and 2018 search results as background dataset. We know the summer season is between June and March, the normal flu season is other months. We will observe variances of those symptoms in summer season and in normal flu season.

Before March 2020 people were not searching COVID-19 symptoms at summer season. People were searching more about those symptoms at summer season after the outbreak of corona virus after March 2020. As people were affected with the corona virus, people did search those symptoms to understand about their health condition. If we apply our algorithm for overall USA data and individually on every state, the variance of COVID-19 symptoms will be higher at summer season. The serial of those 3 symptoms will appear in first second and third position of 11 position for total 11 symptoms. In flu season, both diseases show these 11 symptoms. Before March 2020 and after March 2020, both flu seasons contain similar information of those symptoms. Variations of those symptoms will not maintain the order at normal flu season.

3.4.1 Symptoms strength for overall USA in June

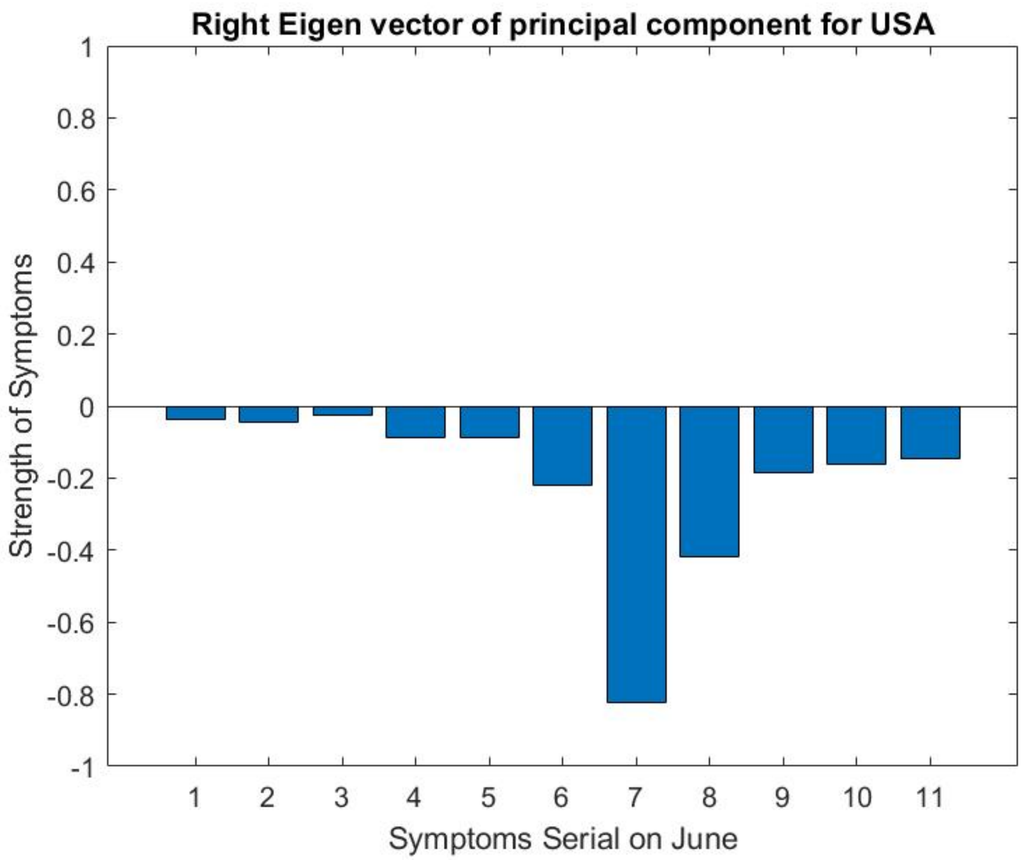

Figure 5: Coefficients for USA in June 2018

This dataset represents single vector column of our right singular vector matrix for 11 symptoms of COVID-19 and seasonal flu. All bar graph values are coefficients of 11 symptoms according to the principle singular value of our diagonal matrix of singular values. First three values are for COVID-19 symptoms of Ageusia, Shortness of Breath, and Anosmia. Values of coefficients of those three symptoms are higher than that of the rest other symptoms at summer season for overall USA.

3.4.2 Symptoms strength for overall USA in March

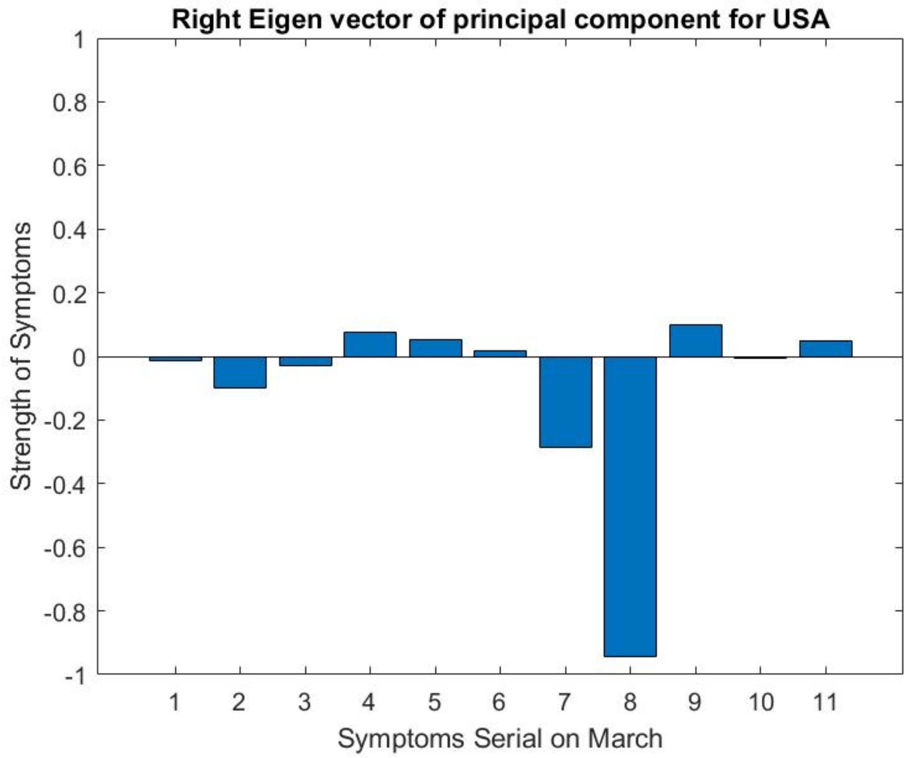

Figure 6: Coefficients for USA in March 2018

This dataset represents single vector column of our right singular vector matrix for 11 symptoms of COVID-19 and seasonal flu. All bar graph values are coefficients of 11 symptoms according to the principle singular value of our diagonal matrix of singular values. First three values are for COVID-19 symptoms of Ageusia, Shortness of Breath, and Anosmia. We can see values of coefficients of those three symptoms are not only higher values at normal flu season for overall USA.

3.4.3 Symptoms strength for Florida in June

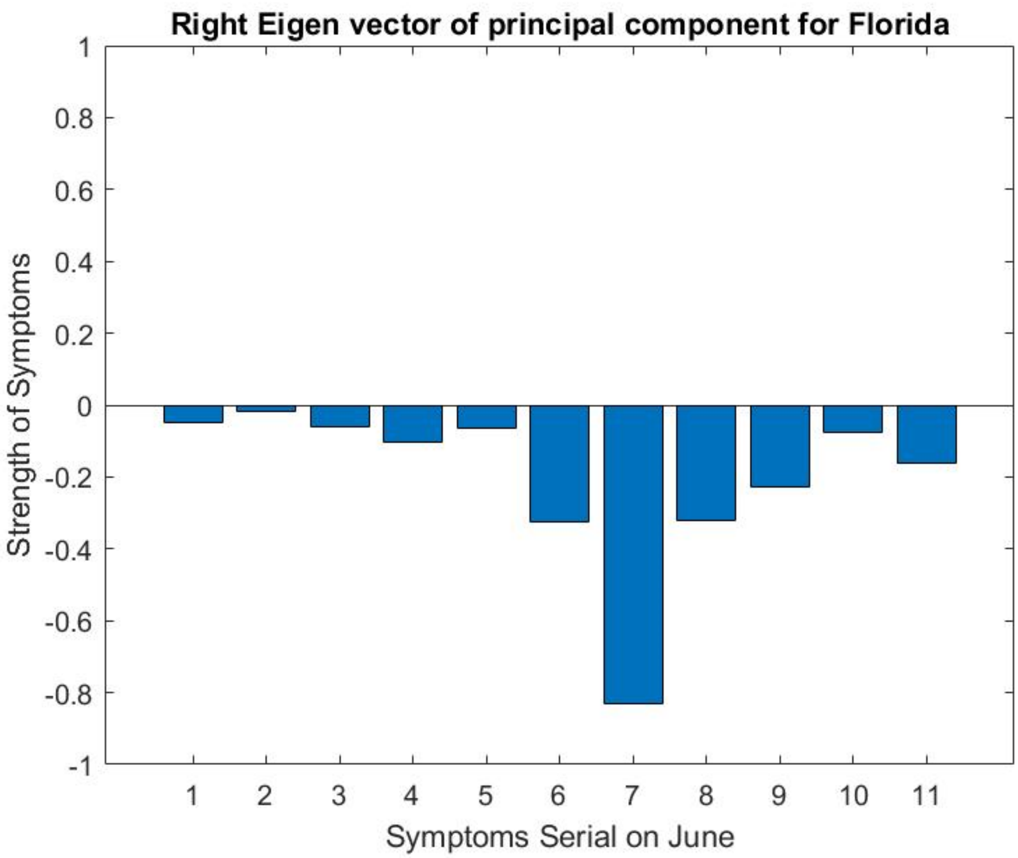

Figure 7: Coefficients for Florida in June 2018

This dataset represents single vector column of our right singular vector matrix for 11 symptoms of COVID-19 and seasonal flu. All bar graph values are coefficients of 11 symptoms according to the principle singular value of our diagonal matrix of singular values. First three values are for COVID-19 symptoms of Ageusia, Shortness of Breath, and Anosmia. Values of coefficients of those three symptoms are higher than that of the rest other symptoms at summer season for the single state Florida.

3.4.4 Symptoms strength for Florida in March

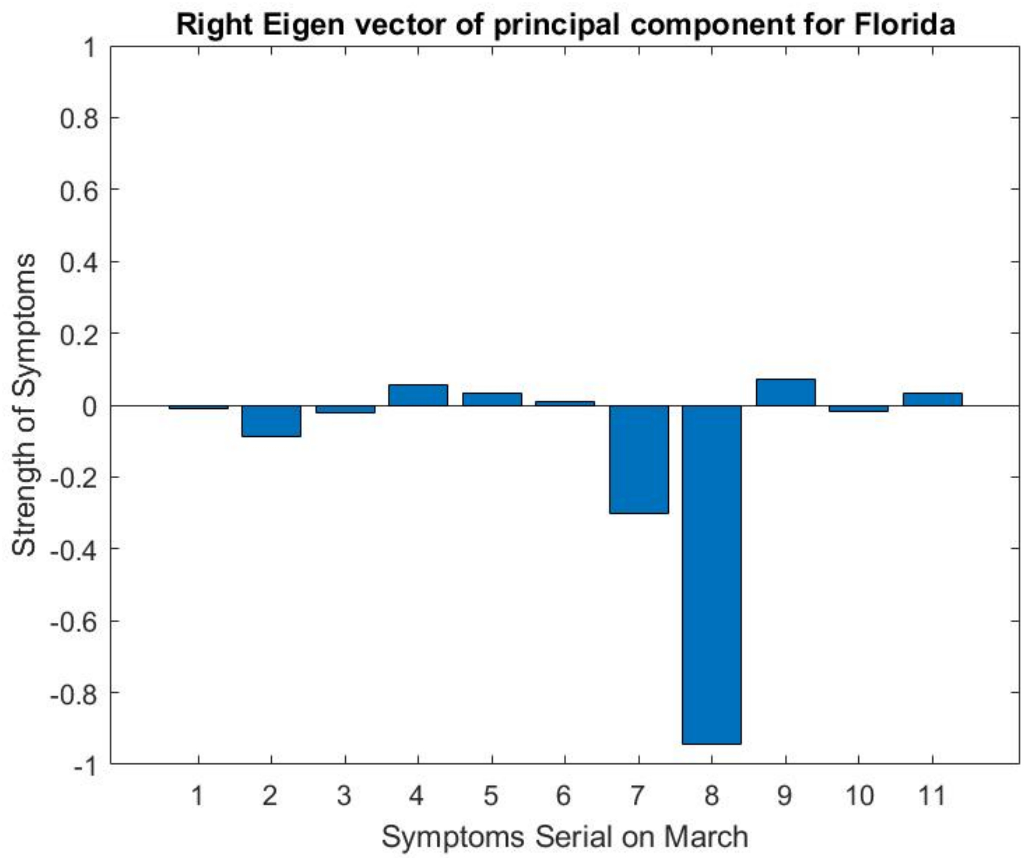

Figure 8: Coefficients for Florida in March 2018

This dataset represents single vector column of our right singular vector matrix for 11 symptoms of COVID-19 and seasonal flu. All bar graph values are coefficients of 11 symptoms according to the principle singular value of our diagonal matrix of singular values. First three values are for COVID-19 symptoms of Ageusia, Shortness of Breath, and Anosmia. We can see values of coefficients of those three symptoms are not only higher values at normal flu season for the single state Florida.

3.4.5 Symptoms list for 6 months for USA

Table 4: Coefficients Position for USA in 2018

	Serial of Symptoms
June	2 1 3 5 10 4 11 9 8 6 7
July	1 3 2 5 4 9 6 10 7 11 8
August	3 1 2 5 4 10 7 6 11 9 8
December	11 9 10 2 6 4 8 5 7 1 3
January	7 11 6 8 9 4 10 5 2 1 3
February	8 7 11 2 3 5 1 9 6 4 10

We have applied our algorithm for overall USA dataset for 6 months period. The above list represents position of our 11 symptoms when background dataset is 2019 search result, and the target dataset is 2020 search result. All positions in the above list are ordered in highest to lowest from left to right. Numbers are for columns in which we have assigned our 11 symptoms. In list, 1, 2 and 3 stands for symptoms Ageusia, Shortness of Breath, and Anosmia. June, July and August, these three months are summer season and December, January, February are normal flu season. Since the variance of COVID-19 symptoms are higher in summer season, these three symptoms have appeared in the 1^{st}, 2^{nd}, and 3^{rd} position in the summer season. In normal flu season, the appearance of those symptoms does not maintain the order because their variances are not comparatively that much higher than that of other symptoms.

3.4.6 Symptoms list for 12 months for Florida

Table 5: Coefficients Position for Florida in 2018

	Serial of Symptoms
June	1 3 2 5 4 10 11 9 6 8 7
July	1 3 2 5 4 6 9 11 10 7 8
August	1 3 2 5 4 9 11 7 10 6 8
December	11 9 6 2 10 4 5 1 3 8 7
January	7 8 6 4 9 11 10 5 2 1 3
February	7 11 4 8 6 9 2 5 3 1 10

We have applied our algorithm for a particular state Florida dataset for 6 months period. The above list represents position of our 11 symptoms when background dataset is 2019 search result, and the target dataset is 2020 search result. All positions in the above list are ordered in highest to lowest from left to right. Numbers are for columns in which we have assigned our 11 symptoms. In list, 1, 2 and 3 stands for symptoms Ageusia, Shortness of Breath, and Anosmia. June, July and August, these three months are summer season and December, January, February are normal flu season. Since the variance of COVID-19 symptoms are higher in summer season, these three symptoms have appeared in the 1st, 2nd, and 3rd position in the summer season. In normal flu season, the appearance of those symptoms does not maintain the order because their variances are not comparatively that much higher than that of other symptoms.

3.5 Results for target data 2019 and background data 2020

In this case we will alter background and target datasets. We will consider 2020 search result as background datasets and 2019 search results as target datasets. So, background dataset will contain higher information for corona virus symptoms. And the searching frequency for corona virus symptoms will be less visible in target dataset. When we had considered 2020 search result as target dataset and 2019 search result as background dataset, variations of COVID-19 symptoms were higher than that of rest symptoms. But for this case the variation will not maintain the order. We will apply our algorithm for overall USA search results and for a single state. Both cases, we will expect the same result. We know the summer season is between June and March, the normal flu season is other months. We will observe variances of those symptoms in summer season and in normal flu season.

If we apply our algorithm in this way, we will not be able to extract meaningful information from the output. Though we will not get meaningful information after implementing this experiment, but we are testing because of verification that our algorithm is properly working. We will apply our algorithm for summer season only here. We will see results for overall USA data and for a single state. The variation of those corona virus symptoms will not maintain the order both cases.

3.5.1 Symptoms strength for overall USA in June

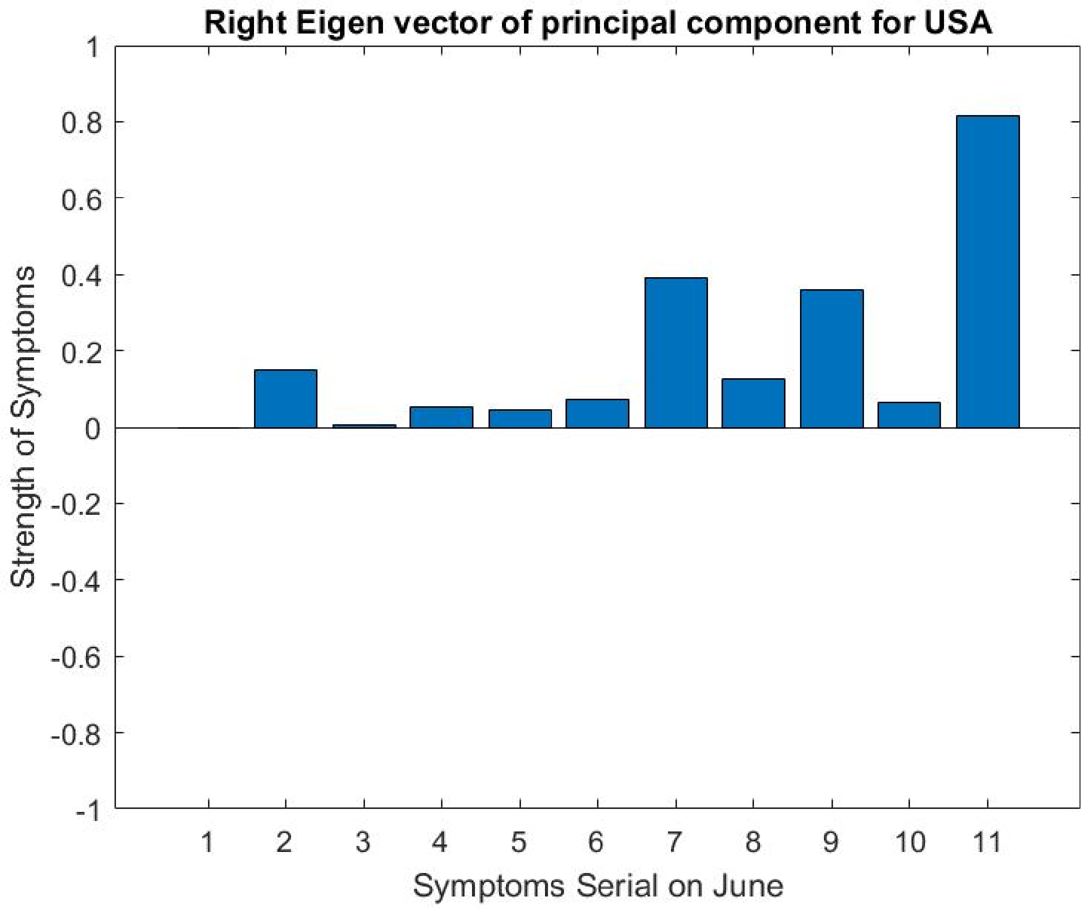

Figure 9: Coefficients for USA in June 2020

This dataset represents single vector column of our right singular vector matrix for 11 symptoms of COVID-19 and seasonal flu. All bar graph values are coefficients of 11 symptoms according to the principle singular value of our diagonal matrix of singular values. First three values are for COVID-19 symptoms of Ageusia, Shortness of Breath, and Anosmia. We can see values of coefficients of those three symptoms are not only higher values at normal flu season for overall USA.

3.5.2 Symptoms strength for Florida in June

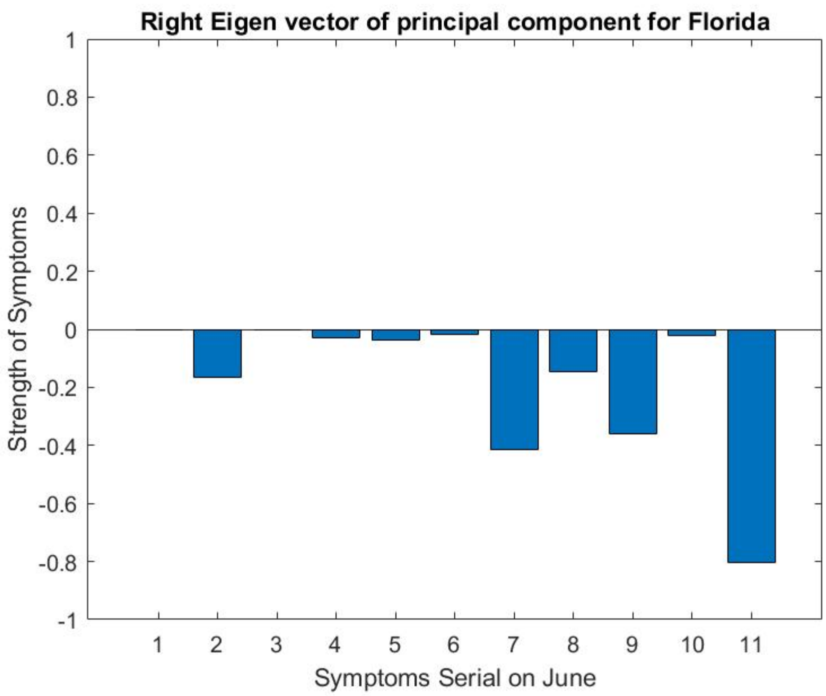

Figure 10: Coefficients for Florida in June 2020

This dataset represents single vector column of our right singular vector matrix for 11 symptoms of COVID-19 and seasonal flu. All bar graph values are coefficients of 11 symptoms according to the principle singular value of our diagonal matrix of singular values. First three values are for COVID-19 symptoms of Ageusia, Shortness of Breath, and Anosmia. We can see values of coefficients of those three symptoms are not only higher values at normal flu season for the single state Florida.

3.6 Results for target data 2018 and background data 2020

In this case we will alter background and target datasets. We will consider 2020 search result as background datasets and 2018 search results as target datasets. So, background dataset will contain higher information for corona virus symptoms. And the searching frequency for corona virus symptoms will be less visible in target dataset. When we had considered 2020 search result as target dataset and 2018search result as background dataset, variations of COVID-19 symptoms were higher than that of rest symptoms. But for this case the variation will not maintain the order. We will apply our algorithm for overall USA search results and for a single state. Both cases, we will expect the same result. We know the summer season is between June and March, the normal flu season is other months. We will observe variances of those symptoms in summer season and in normal flu season.

If we apply our algorithm in this way, we will not be able to extract meaningful information from the output. Though we will not get meaningful information after implementing this experiment, but we are testing because of verification that our algorithm is properly working. We will apply our algorithm for summer season only here. We will see results for overall USA data and for a single state. The variation of those corona virus symptoms will not maintain the order both cases.

3.6.1 Symptoms strength for overall USA in June

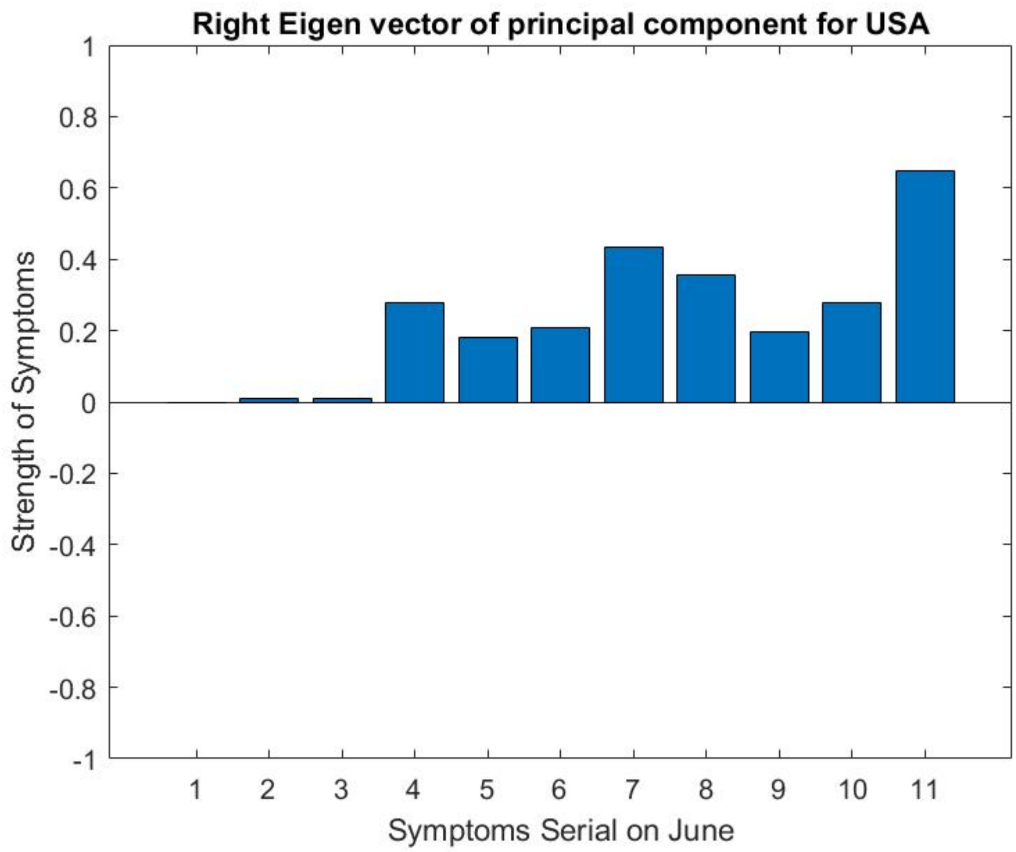

Figure 11: Coefficients for USA in June 2020

This dataset represents single vector column of our right singular vector matrix for 11 symptoms of COVID-19 and seasonal flu. All bar graph values are coefficients of 11 symptoms according to the principle singular value of our diagonal matrix of singular values. First three values are for COVID-19 symptoms of Ageusia, Shortness of Breath, and Anosmia. We can see values of coefficients of those three symptoms are not only higher values at normal flu season for overall USA.

3.6.2 Symptoms strength for Florida in June

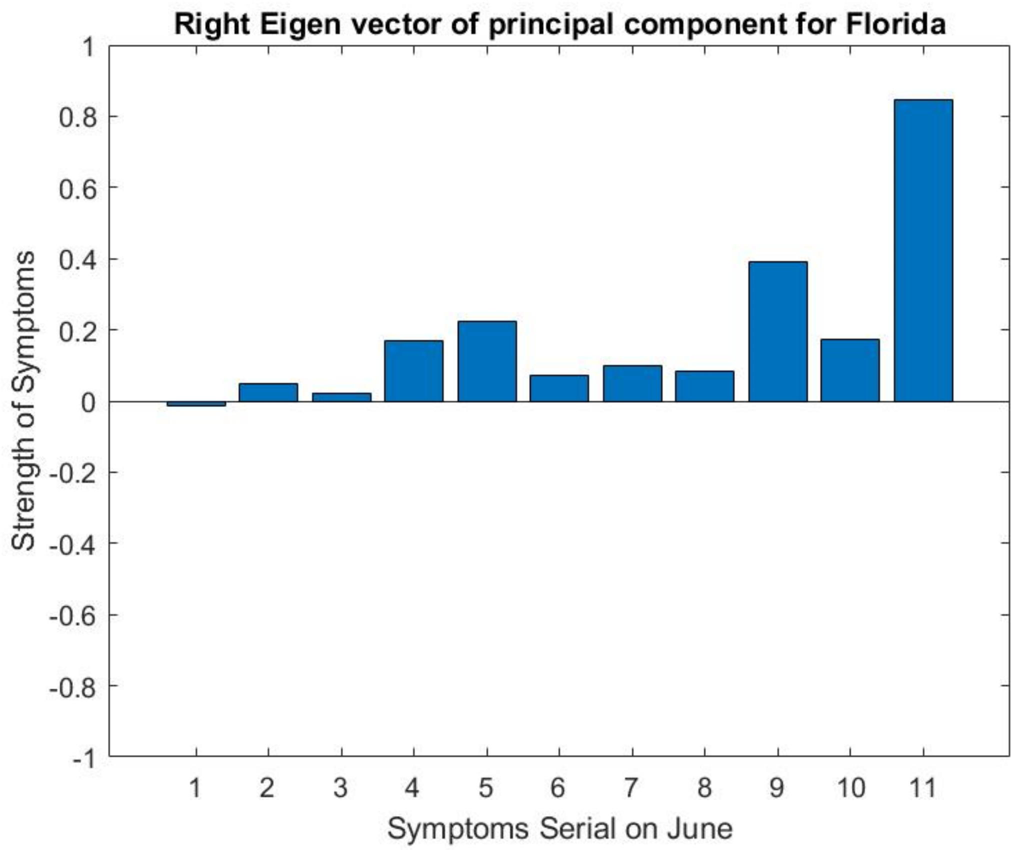

Figure 12: Coefficients for Florida in June 2020

This dataset represents single vector column of our right singular vector matrix for 11 symptoms of COVID-19 and seasonal flu. All bar graph values are coefficients of 11 symptoms according to the principle singular value of our diagonal matrix of singular values. First three values are for COVID-19 symptoms of Ageusia, Shortness of Breath, and Anosmia. We can see values of coefficients of those three symptoms are not only higher values at normal flu season for the single state Florida.

CHAPTER IV

SUMMARY AND CONCLUSION

We have developed an algorithm considering mathematical concepts of different types of matrix factorization techniques. These principles, we have briefly described in our literature review section. We have applied this algorithm on online open-source search result of different illness symptoms to find out discriminative information between COVID-19 and seasonal flu symptoms. Before applying our algorithm, we have considered two assumption from visualizing target and background datasets. The first assumption is that target dataset contains higher frequency of search results for corona virus symptoms at summer season than that of background datasets. The variation of COVID-19 symptoms will be higher in summer season not only on the overall search result for USA but also for search results of a single state. The second assumption is that both diseases demonstrate those 11 symptoms in the normal flu season. Coefficients of those symptoms will maintain the order in normal flu season.

We have applied our algorithm for three cases. In first case, we have considered 2020 search result for target dataset and 2019 search result as background dataset. In second case, we also have considered 2020 search result as target dataset, but the background dataset is now 2018 search result. In third case, we have altered the order of our datasets. We have considered 2018 or 2019 search result as target dataset and the 2020 search result as background dataset.

We have got our desired output for three cases. Variations of COVID-19 symptoms were higher in summer season for first two cases. In flu season symptoms do not maintain the order what we have expected before applying our algorithm. We also have got desired result in case of third case. Variations of those symptoms do not maintain order here. We have seen these three symptoms have taken first three position at summer season.

REFERENCES

[1] Abubakar Abid, Martin J Zhang, Vivek K Bagaria, and James Zou. 2017. Contrastive principal component analysis. arXiv preprint arXiv:1709.06716 (2017).

[2] Seyed Mohammad Ayyoubzadeh, Seyed Mehdi Ayyoubzadeh, Hoda Zahedi, Mahnaz Ahmadi, and Sharareh R Niakan Kalhori. 2020. Predicting COVID-19 incidence through analysis of google trends data in iran: data mining and deep learning pilot study. JMIR public health and surveillance 6, 2 (2020), e18828.

[3] Jia Chen, Gang Wang, and Georgios B Giannakis. 2018. Nonlinear dimensionality reduction for discriminative analytics of multiple datasets. IEEE Transactions on Signal Processing 67, 3 (2018), 740–752.

[4] Maria Effenberger, Andreas Kronbichler, Jae Il Shin, Gert Mayer, Herbert Tilg, and Paul Perco. 2020. Association of the COVID-19 pandemic with internet search volumes: a Google TrendsTM analysis. International Journal of Infectious Diseases 95 (2020), 192–197.

[5] Atina Husnayain, Anis Fuad, and Emily Chia-Yu Su. 2020. Applications of Google Search Trends for risk communication in infectious disease management: A case study of the COVID-19 outbreak in Taiwan. International Journal of Infectious Diseases 95 (2020),

221–223.

[6] Ariel Jaffe and Mati Wax. 2014. Single-site localization via maximum discrimination multipath fingerprinting. IEEE Transactions on Signal Processing 62, 7 (2014), 1718–1728.

[7] D Lee and HS Seung. 2000. Algorithms for non˜ negative matrix factorization. In Proc. of the 13th Int. Conf on Neural Information Processing Systems (NeurIPS 2000). New York: CurranAssociates, Vol. 535541.

[8] Cuilian Li, Li Jia Chen, Xueyu Chen, Mingzhi Zhang, Chi Pui Pang, and Haoyu Chen. 2020. Retrospective analysis of the possibility of predicting the COVID-19 outbreak from Internet searches and social media data, China, 2020. Eurosurveil- lance 25, 10 (2020), 2000199.

[9] Google LLC. 2021. Google COVID-19 Search Trends symptoms dataset. http: //goo.gle/covid19symptomdataset. (Feb. 2021).

[10] Amaryllis Mavragani. 2020. Tracking COVID-19 in Europe: infodemiology approach. JMIR public health and surveillance 6, 2 (2020), e18941.

[11] Amaryllis Mavragani and Konstantinos Gkillas. 2020. COVID-19 predictability in

the United States using Google Trends time series. Scientific reports 10, 1 (2020), 1–12.

[12] T. G. Kolda, B. W. Bader, and J. P. Kenny, “Higher-order web link analysis using multilinear algebra,” in Fifth IEEE International Conference on Data Mining (ICDM’05), Nov 2005, pp. 8 pp.–.

[13] E. Acar, S.A. Camtepe, M.S. Krishnamoorthy, and B. Yener, Modeling and Multiway Analysis of Chatroom Tensors, pp. 256–268, Springer Berlin Heidelberg, Berlin, Heidelberg, 2005.

[14] E. Acar and B. Yener, “Unsupervised multiway data analysis: A litera- ture survey,” IEEE Transactions on Knowledge and Data Engineering, vol. 21, no. 1, pp. 6–20, Jan 2009.

[15] D. Nion, K.N. Mokios, N.D. Sidiropoulos, and A. Potamianos, “Batch and adaptive PARAFAC-based blind separation of convolutive speech mixtures,” IEEE Transactions on Audio, Speech, and Language Processing, vol. 18, no. 6, pp. 1193–1207, 2010.

[16] N.D. Sidiropoulos, G.B. Giannakis, and R. Bro, “Blind PARAFAC receivers for DS-CDMA systems,” IEEE Transactions on Signal Processing, vol. 48, no. 3, pp. 810–823, 2000.

[17] D. Nion and N.D. Sidiropoulos, “Tensor algebra and multidimensional harmonic retrieval

in signal processing for MIMO radar," IEEE Transactions on Signal Processing, vol. 58, no. 11, pp. 5693–5705, Nov 2010.

[18] N. D. Sidiropoulos, R. Bro, and G. B. Giannakis, "Parallel factor analysis in sensor array processing," IEEE Trans. Signal Process., vol. 48, no. 8, pp. 2377–2388, Aug. 2000.

[19] A.-J. van der Veen and A. Paulraj, "An analytical constant modulus algorithm," IEEE Transactions on Signal Processing, vol. 44, no. 5, pp. 1136–1155, May 1996.

[20] E. Kofidis and P. Regalia, "Tensor approximation and signal processing applications," Contemporary Mathematics, vol. 280, pp. 103–134, 2001.

[21] M. Vasilescu and D. Terzopoulos, Multilinear Analysis of Image Ensembles: TensorFaces, pp. 447–460, Springer Berlin Heidelberg, Berlin, Heidelberg, 2002.

[22] A. Cichocki, D. Mandic, L. De Lathauwer, Zhou Q., Zhao Q., C. Ca- iafa, and A.H. Phan, "Tensor decompositions for signal processing applications: From two-way to multiway component analysis," Signal Processing Magazine, IEEE, vol. 32, no. 2, pp. 145–163, March 2015.

[23] E.E. Papalexakis, C. Faloutsos, and N.D. Sidiropoulos, "Parcube: Sparse parallelizable tensor decompositions.," in ECML/PKDD (1), P.A. Flach, T.D. Bie, and

N. Cristianini, Eds. 2012, vol. 7523 of Lecture Notes in Computer Science, pp. 521–536, Springer.

[24] A. Anandkumar, R. Ge, D. Hsu, S. M. Kakade, and M. Telgarsky, “Tensor decompositions for learning latent variable models,” J. Mach. Learn. Res., vol. 15, no. 1, pp. 2773–2832, Jan. 2014.

[25] T.G. Kolda and B.W. Bader, “Tensor decompositions and applications,” SIAM REVIEW, vol. 51, no. 3, pp. 455–500, 2009.

[26] P. Comon, “Tensors : A brief introduction,” Signal Processing Magazine, IEEE, vol. 31, no. 3, pp. 44–53, May 2014.

[27] L. Grasedyck, D. Kressner, and C. Tobler, “A literature survey of low- rank tensor approximation techniques,” GAMM-Mitteilungen, vol. 36, no. 1, pp. 53–78, 2013.

[28] W. Hackbusch, Tensor spaces and numerical tensor calculus, vol. 42, Springer Science & Business Media, 2012.

[29] N. Vervliet, O. Debals, L. Sorber, and L De Lathauwer, “Breaking the Curse of Dimensionality Using Decompositions of Incomplete Tensors: Tensor-based scientific computing in big data analysis,” Signal Processing Magazine, IEEE, vol. 31, no. 5,

pp. 71–79, Sept. 2014.

[30] R. Bro, “PARAFAC. Tutorial and applications,” Chemometrics and Intelligent Laboratory Systems, vol. 38, pp. 149–171, 1997.

[31] A.K. Smilde, R. Bro, and P. Geladi, Multi-way analysis with applica- tions in the chemical sciences, John Wiley & Sons, 2004.

[32] E.E. Papalexakis, C. Faloutsos, N.D. Sidiropoulos, and T. Kolda, “Ten- sors for data mining and data fusion: Models, applications, and scalable algorithms,” ACM Trans. on Intelligent Systems and Technology, vol. 8, no. 2, pp. 16:1–16:44, Nov. 2016.

BIOGRAPHICAL SKETCH

Md Jakir Hossain completed his Master of Science in Electrical and Computer Engineering from The University of Texas at Rio Grande Valley in December 2021. His research area is Applied Machine Learning. He is interested in doing his PhD in the intersection of Communication, Control, Signal Processing and Machine Learning. His personal email address is jakir311285@gmail.com