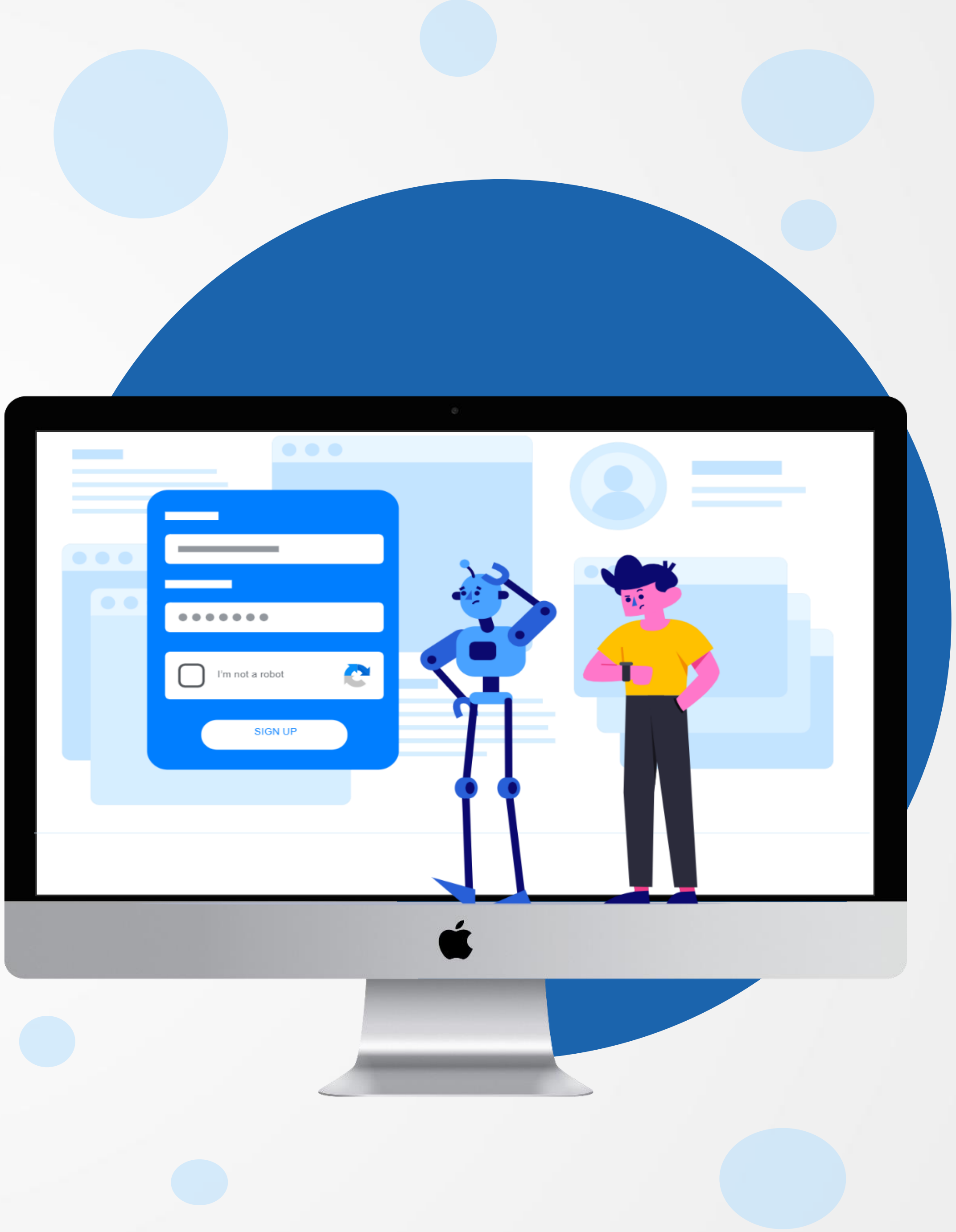
GNG 5125 Data Science Application Group Assignment

# Psychological Chatbot

## Group Six

| | |
|---|---|
| Kun Qin | Student Number: 300188345 |
| Zhixin He | Student Number: 300142825 |
| Wenqi Lin | Student Number: 300201446 |
| Yuxue Pang | Student Number: 300177911 |



# Machine guard mental health!

## Psychological problems in reality

□ Impact of the COVID-19

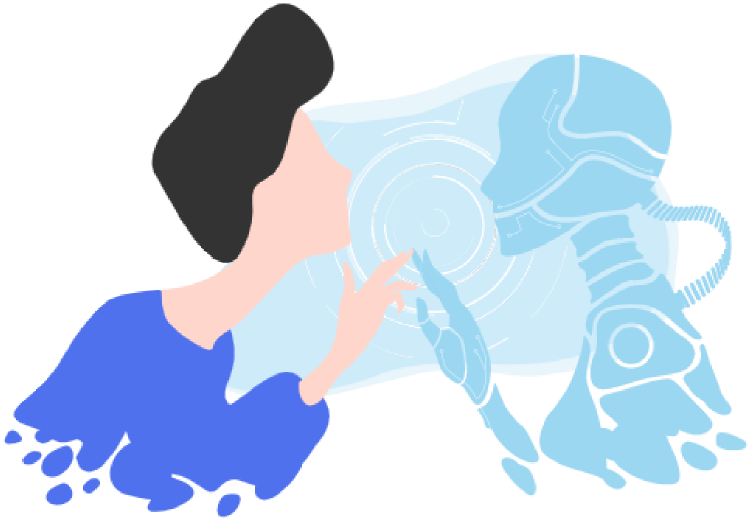
□ Increasing pressure

□ Don't know where and who can help

□ No response when needed

## Psychological Chatbot

- Professional psychology knowledge
- Providing service whenever you need it
- Simple operation page
- The recommendation system considers your needs



### 1. Model

Prepare data
Use data to train the model

### 2. Answer

User input question
The model classifies the question
Output the corresponding question

### 3. Recommend

Confusion matrix find similar problems
Recommend related questions to Unser

### 4. Feedback

User clicks on question links
Return the answer to users

# Content

1

## Data

Preparation Process Transform

2

## Algorithm

Classification Clustering

3

## Analysis

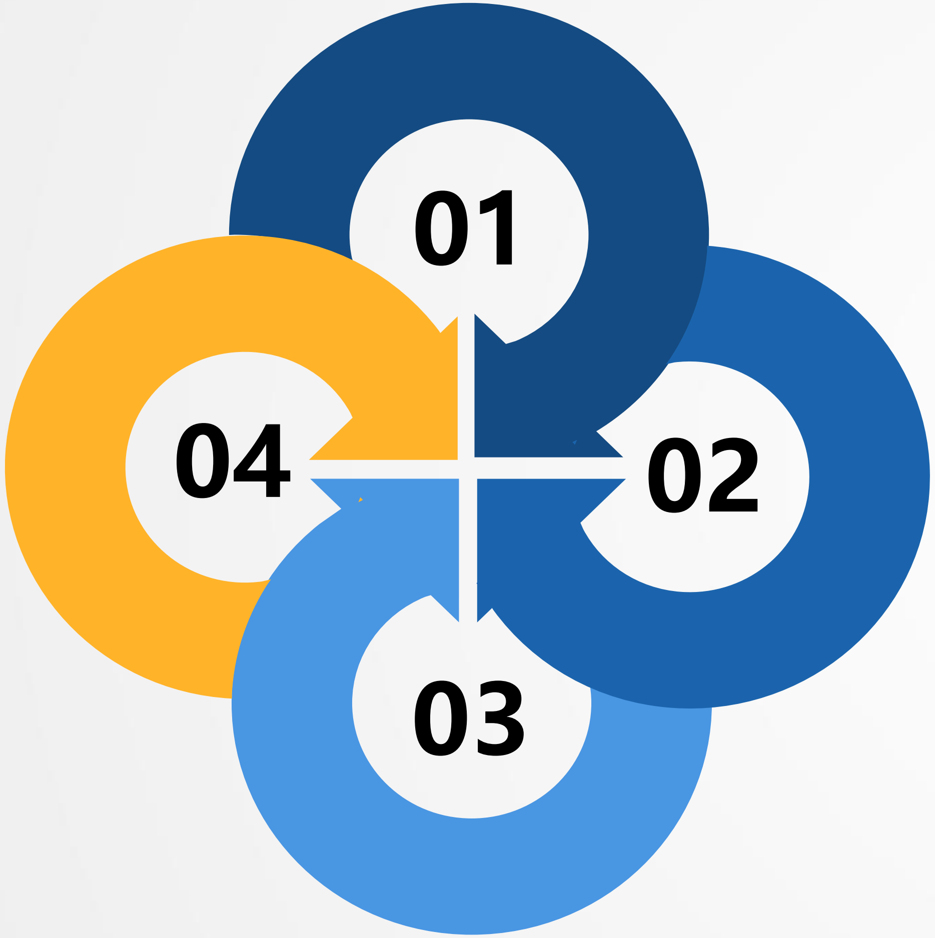
Accuracy Confusion Matrix

4

## Innovation

Platform Recommended system

# Data



## Preparation

- ■ Download dataset form website Kaggle
- ■ **98** professional questions with answers

## Enrich

- □ Adding **4** other questions for original questions
- □ Each group get **5** questions and **1** answer

## Clean

- ■ Clean up stop words and punctuation in the problem
- ■ Clean up special characters in answers

## Transform

- □ Bag of words
- □ Term frequency–inverse document frequency

https://www.kaggle.com/narendrageek/mental-health-faq-for-chatbot

# Classification

| Algorithm | BOW | TF-IDF |
|---|---|---|
| SVM | 0.79 | 0.83 |
| KNN | 0.73 | 0.75 |
| DC | 0.75 | 0.71 |

| | |
|---|---|
| SVM + BOW | svm.SVC(C=0.5, kernel='linear', decision_function_shape='ovo') |
| SVM + TF-IDF | svm.SVC(kernel='linear', C=1) |
| KNN + BOW | KNeighborsClassifier(n_neighbors=1,algorithm='kd_tree',weights = 'distance') |
| KNN + TF-IDF | KNeighborsClassifier(n_neighbors=2,algorithm='auto',weights = 'distance', p = 3) |
| DC + BOW | tree.DecisionTreeClassifier(splitter='best', criterion = 'gini') |
| DC + TF-IDF | tree.DecisionTreeClassifier(splitter='best') |

# Clustering

| Algorithm | | Kappa | Consistency | silScore | chScore |
|---|---|---|---|---|---|
| K-Means | BOW | 0.041237 | 0.182656 | 0.066978 | 3.206853 |
| | TF-IDF | 0.041237 | 0.249760 | 0.030693 | 3.953466 |
| GM | BOW | 0.041237 | 0.284635 | 0.066765 | 3.178924 |
| | TF-IDF | 0.041237 | 0.270914 | 0.029805 | 5.206064 |
| HC | BOW | 0.047423 | 0.306309 | 0.017640 | 2.590824 |
| | TF-IDF | 0.051546 | 0.276610 | 0.052813 | 3.191267 |

**K-Means**

```
kmeans = KMeans(n_clusters = 5, init='k-means++', n_init=10, max_iter=400,
                tol=0.0001, precompute_distances=True, verbose=0, random_state=9,
                copy_x=True, n_jobs=None, algorithm='elkan')
```

**GM**

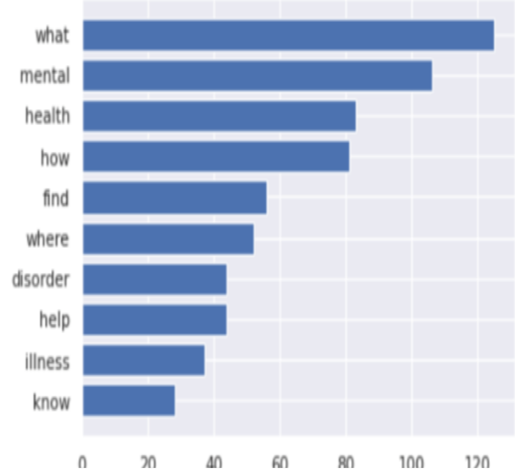
```
gmm = GaussianMixture(n_components=5, covariance_type='full', tol=0.0001,
reg_covar=1e-06, max_iter=300, n_init=200,weights_init=None, means_init=None,
precisions_init=None, random_state=None, warm_start=False,
                verbose=0, verbose_interval=10)
```

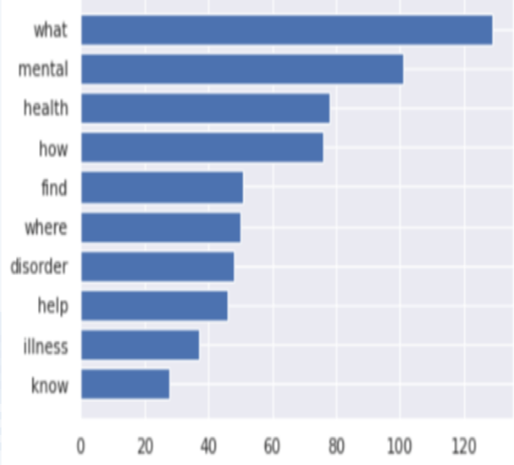
**HC**

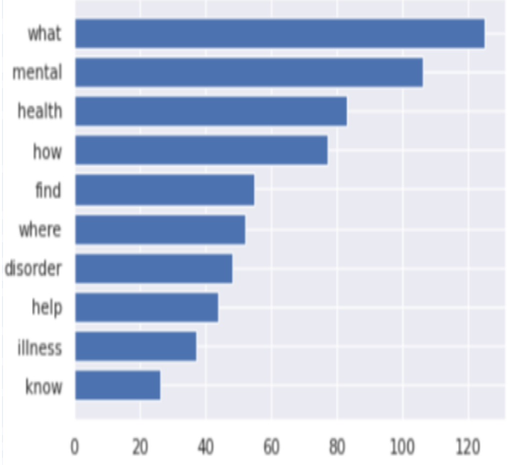
```
ac = AgglomerativeClustering(n_clusters=5, affinity='euclidean', linkage='ward')
```

# Analysis

Frequency Words through BOW





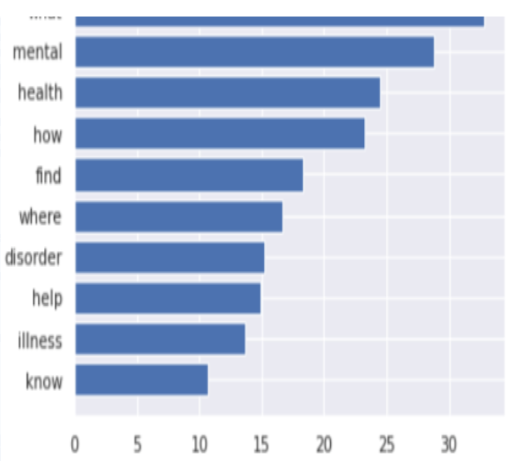


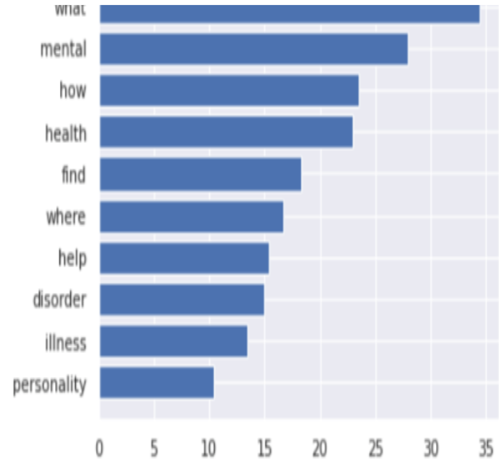
KMeans

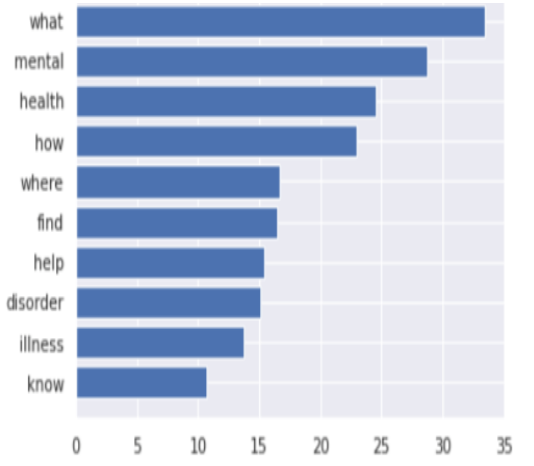
EM

HC

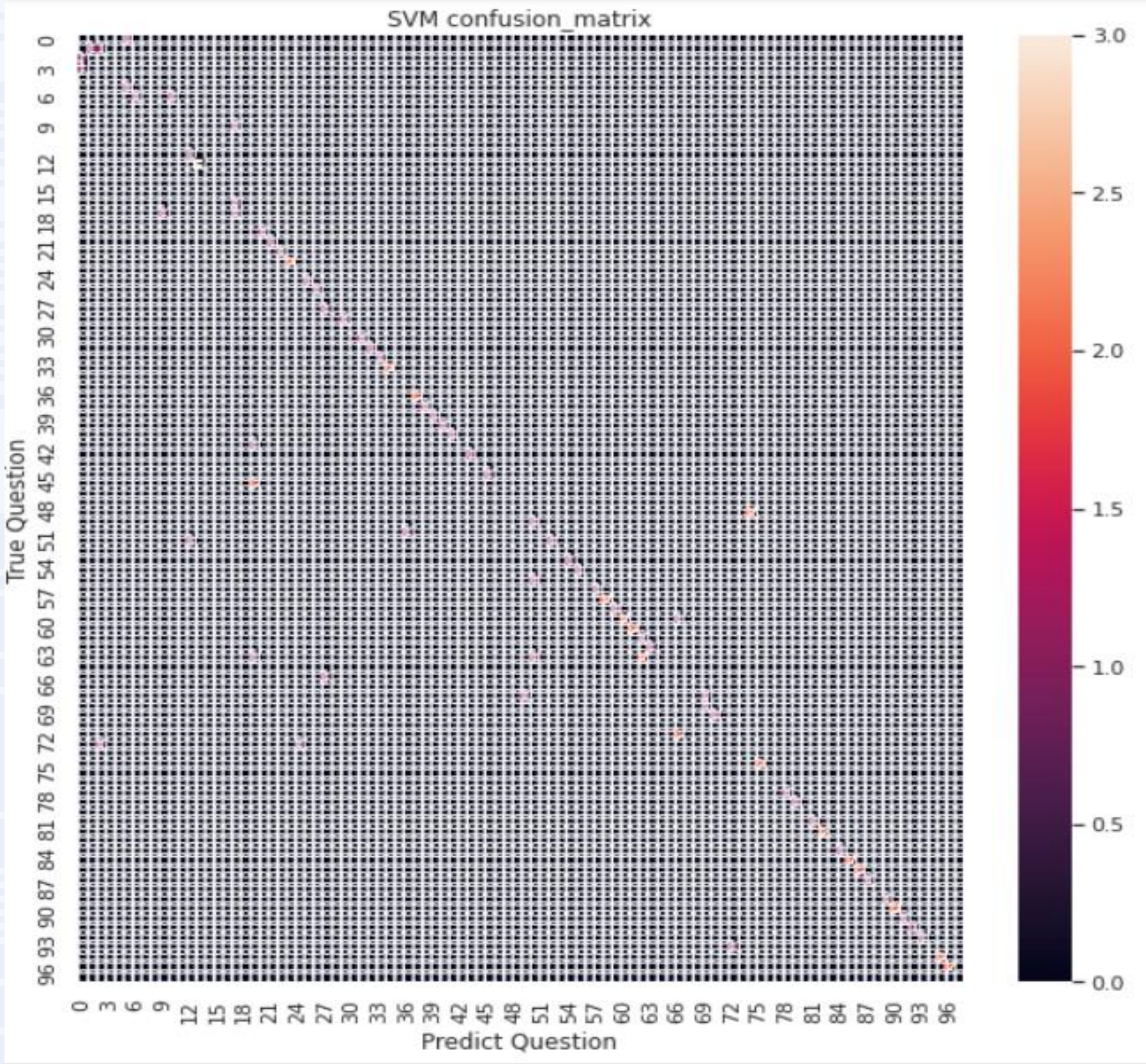
Frequency Words through TD-IDF







# Analysis



## TF-IDF + SVM

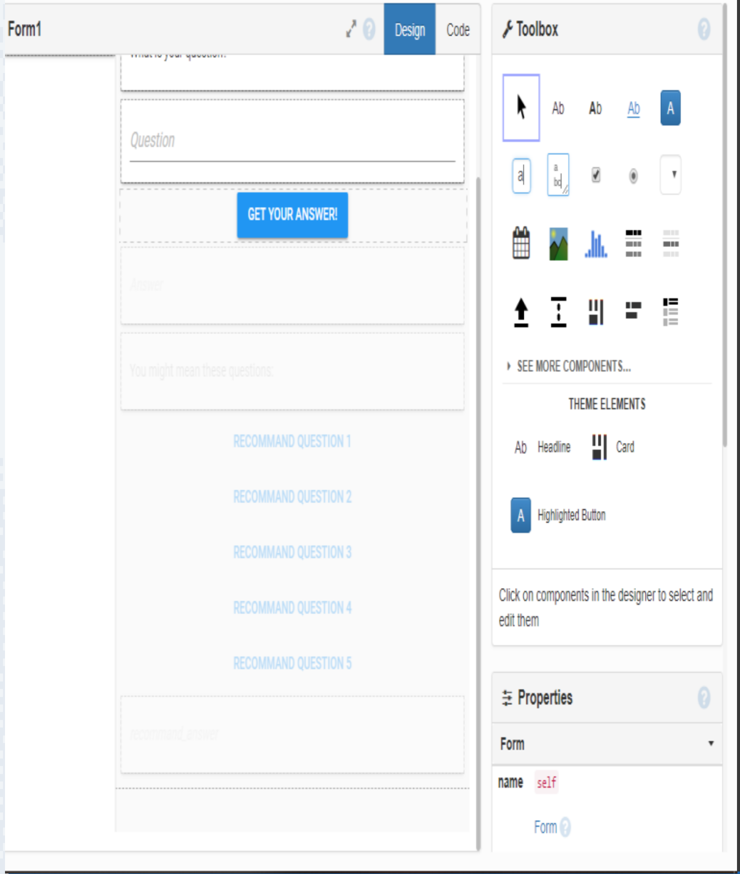
The machine has some problems in distinguishing between 66 to 72 and 12 to 24, which means that these problem groups may be somewhat similar to other problems.

The distribution of the model looks like a diagonal line from the upper left corner to the lower right corner.

The model we design has good accuracy most of the time.

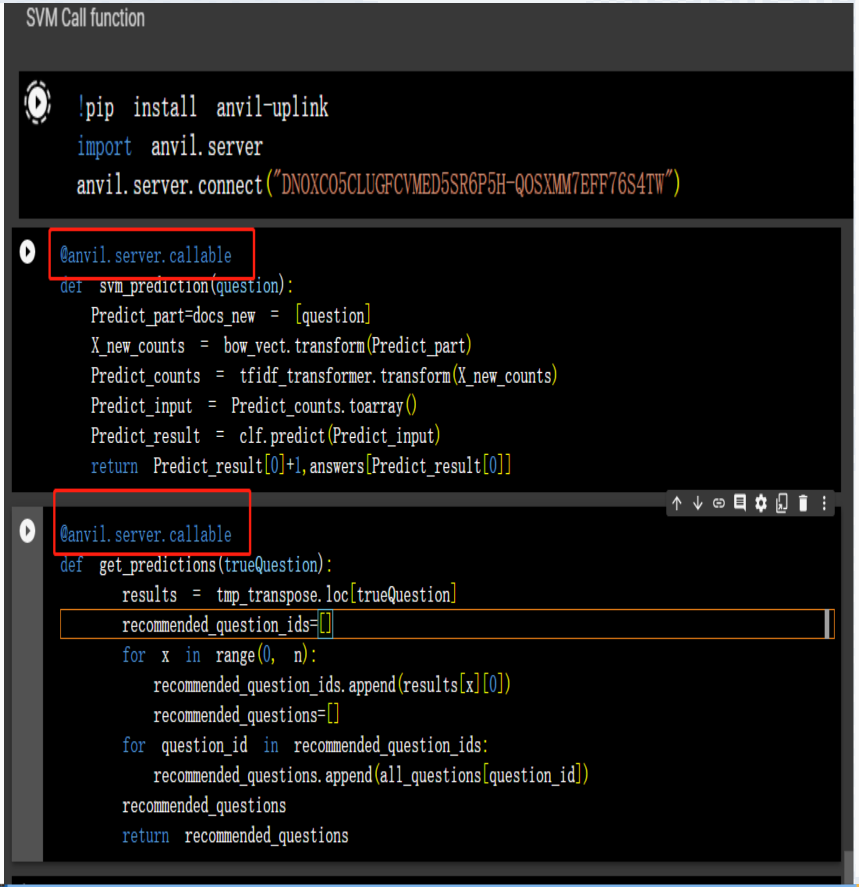
# Innovation

Our group used the anvil online platform to realize the GUI development. Anvil is a platform that can be connect with google colab to do amazing front end display.



Click on components in the designer to select and edit them

Design Page



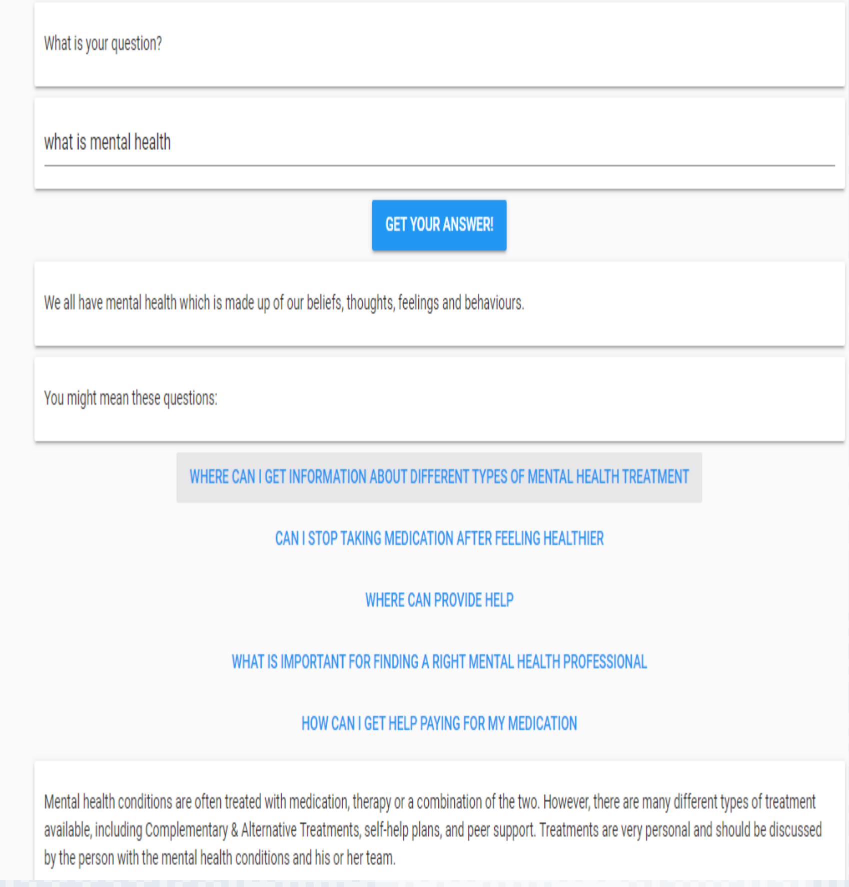
SVM Call function

```
!pip install anvil-uplink
import anvil.server
anvil.server.connect("DNOXCO5CLUGFCVMED5SR6P5H-QOSXMM7EFF76S4TW")
```

```
@anvil.server.callable
def svm_prediction(question):
    Predict_part=docs_new = [question]
    X_new_counts = bow_vect.transform(Predict_part)
    Predict_counts = tfidf_transformer.transform(X_new_counts)
    Predict_input = Predict_counts.toarray()
    Predict_result = clf.predict(Predict_input)
    return Predict_result[0]+1,answers[Predict_result[0]]
```

```
@anvil.server.callable
def get_predictions(trueQuestion):
        results = tmp_transpose.loc[trueQuestion]
        recommended_question_ids=[]
        for x in range(0, n):
            recommended_question_ids.append(results[x][0])
            recommended_questions=[]
        for question_id in recommended_question_ids:
            recommended_questions.append(all_questions[question_id])
        recommended_questions
        return recommended_questions
```

Connect Colab



What is your question?

what is mental health

GET YOUR ANSWER!

We all have mental health which is made up of our beliefs, thoughts, feelings and behaviours.

You might mean these questions:

WHERE CAN I GET INFORMATION ABOUT DIFFERENT TYPES OF MENTAL HEALTH TREATMENT

CAN I STOP TAKING MEDICATION AFTER FEELING HEALTHIER

WHERE CAN PROVIDE HELP

WHAT IS IMPORTANT FOR FINDING A RIGHT MENTAL HEALTH PROFESSIONAL

HOW CAN I GET HELP PAYING FOR MY MEDICATION

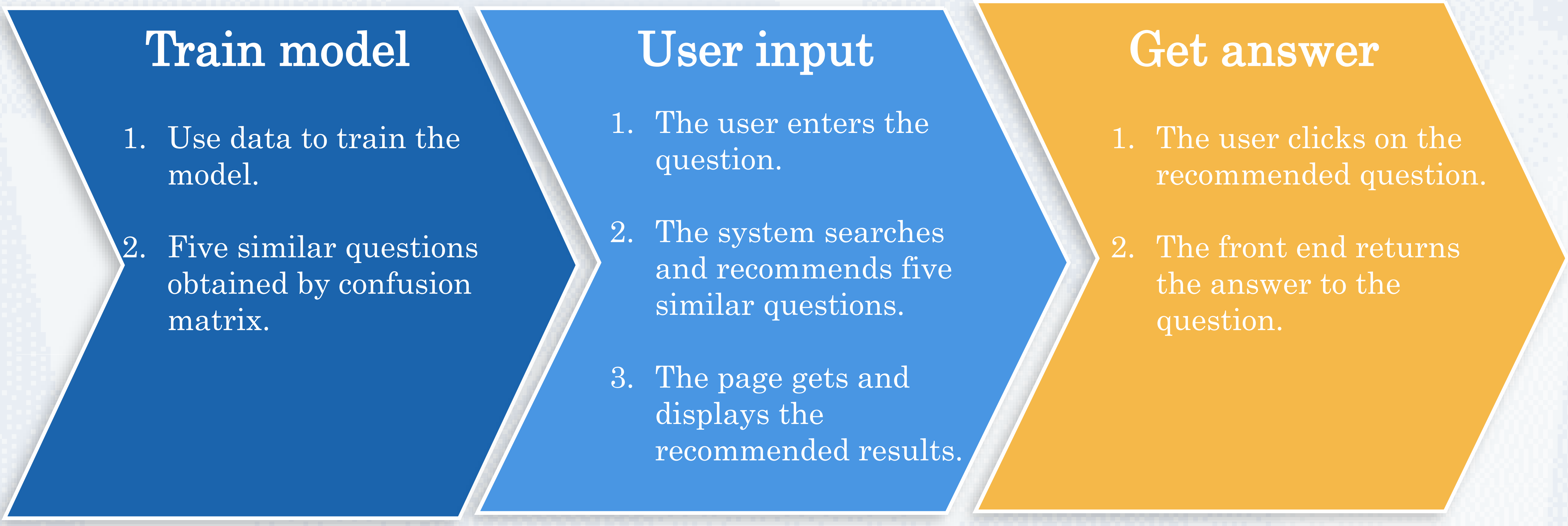
Mental health conditions are often treated with medication, therapy or a combination of the two. However, there are many different types of treatment available, including Complementary & Alternative Treatments, self-help plans, and peer support. Treatments are very personal and should be discussed by the person with the mental health conditions and his or her team.

Final display

# Innovation

When customers ask questions, we will also recommend similar questions.

We obtain similar problems through confusion matrix.



## Train model

1. Use data to train the model.
2. Five similar questions obtained by confusion matrix.

## User input

1. The user enters the question.
2. The system searches and recommends five similar questions.
3. The page gets and displays the recommended results.

## Get answer

1. The user clicks on the recommended question.
2. The front end returns the answer to the question.

# Thank you for listening!

G
G
g

## Waiting for questions!

G
G
g