```
import nltk
import json
from nltk.corpus import wordnet
from nltk.stem import WordNetLemmatizer
import numpy as np
import torch
import torch.nn as nn
import torch.optim as optim
from torch.utils.data import Dataset, DataLoader
import random
import warnings
warnings.filterwarnings('ignore')

nltk.download('punkt')
nltk.download('wordnet')
```

```
[nltk_data] Downloading package punkt to /root/nltk_data...
[nltk_data]   Package punkt is already up-to-date!
[nltk_data] Downloading package wordnet to /root/nltk_data...
[nltk_data]   Package wordnet is already up-to-date!
```

```
True
```

```
with open('intents.json', 'r') as f:
    intents = json.load(f)

intents
```

```
{'intents': [{'tag': 'greeting',
   'patterns': ['Hi',
    'Hey',
    'Is anyone there?',
    'Hi there',
    'Hello',
    'Hey there',
    'Howdy',
    'Hola',
    'Bonjour',
    'Konnichiwa',
    'Guten tag',
    'Ola'],
   'responses': ['Hello there. Tell me how are you feeling today?',
    'Hi there. What brings you here today?',
    'Hi there. How are you feeling today?',
    'Great to see you. How do you feel currently?',
    "Hello there. Glad to see you're back. What's going on in your
world right now?"]},
  {'tag': 'morning',
   'patterns': ['Good morning'],
   'responses': ["Good morning. I hope you had a good night's sleep.
How are you feeling today? "]},
```

```
  {'tag': 'afternoon',
   'patterns': ['Good afternoon'],
   'responses': ['Good afternoon. How is your day going?']},
  {'tag': 'evening',
   'patterns': ['Good evening'],
   'responses': ['Good evening. How has your day been?']},
  {'tag': 'night',
   'patterns': ['Good night'],
   'responses': ['Good night. Get some proper sleep',
    'Good night. Sweet dreams.']},
  {'tag': 'goodbye',
   'patterns': ['Bye',
    'See you later',
    'Goodbye',
    'Au revoir',
    'Sayonara',
    'ok bye',
    'Bye then',
    'Fare thee well'],
   'responses': ['See you later.',
    'Have a nice day.',
    'Bye! Come back again.',
    "I'll see you soon."]},
  {'tag': 'thanks',
   'patterns': ['Thanks',
    'Thank you',
    "That's helpful",
    'Thanks for the help',
    'Than you very much'],
   'responses': ['Happy to help!',
    'Any time!',
    'My pleasure',
    "You're most welcome!"]},
  {'tag': 'no-response',
   'patterns': [''],
   'responses': ["Sorry, I didn't understand you.",
    'Please go on.',
    'Not sure I understand that.',
    "Please don't hesitate to talk to me."]},
  {'tag': 'neutral-response',
   'patterns': ['nothing much'],
   'responses': ['Oh I see. Do you want to talk about something?']},
  {'tag': 'about',
   'patterns': ['Who are you?',
    'What are you?',
    'Who you are?',
    'Tell me more about yourself.',
    'What is your name?',
    'What should I call you?',
```

```
    "What's your name?",
    'Tell me about yourself'],
   'responses': ["I'm Pandora, your Personal Therapeutic AI Assistant.
How are you feeling today",
    "I'm Pandora, a Therapeutic AI Assitant designed to assist you.
Tell me about yourself.",
    "I'm Pandora. I am a conversational agent designed to mimic a
therapist. So how are you feeling today?",
    'You can call me Pandora.',
    "I'm Pandora!",
    'Call me Pandora']},
  {'tag': 'skill',
   'patterns': ['What can you do?'],
   'responses': ["I can provide general advice regarding anxiety and
depression, answer questions related to mental health and make daily
conversations. Do not consider me as a subsitute for an actual mental
healthcare worker. Please seek help if you don't feel satisfied with
me."]},
  {'tag': 'creation',
   'patterns': ['Who created you?',
    'How were you made?',
    'How were you created?'],
   'responses': ['I was created by >.',
    'I was trained on a text dataset using Deep Learning & Natural
Language Processing techniques',
    'The real question is: Who created you?']},
  {'tag': 'name',
   'patterns': ['My name is ', 'I am name.', 'I go by '],
   'responses': ['Oh nice to meet you. Tell me how was your week?',
    'Nice to meet you. So tell me. How do you feel today?',
    "That's a great name. Tell me more about yourself."]},
  {'tag': 'help',
   'patterns': ['Could you help me?',
    'give me a hand please',
    'Can you help?',
    'What can you do for me?',
    'I need support',
    'I need help',
    'Support me please'],
   'responses': ['Sure. Tell me how can i assist you',
    'Tell me your problem so that i can assist you',
    'Yes, sure. How can I help you?']},
  {'tag': 'sad',
   'patterns': ['I am feeling lonely',
    'I am so lonely',
    'I feel down',
    'I feel sad',
    'I am sad',
    'I feel so lonely',
```

```
    'I feel empty',
    "I don't have anyone"],
   'responses': ["I'm sorry to hear that. I'm here for you. Talking
about it might help. So, tell me why do you think you're feeling this
way?",
    "I'm here for you. Could you tell me why you're feeling this
way?",
    'Why do you think you feel this way?',
    'How long have you been feeling this way?']},
  {'tag': 'stressed',
   'patterns': ['I am so stressed out',
    'I am so stressed',
    'I feel stuck',
    'I still feel stressed',
    'I am so burned out'],
   'responses': ['What do you think is causing this?',
    'Take a deep breath and gather your thoughts. Go take a walk if
possible. Stay hydrated',
    'Give yourself a break. Go easy on yourself.',
    'I am sorry to hear that. What is the reason behind this?']},
  {'tag': 'worthless',
   'patterns': ['I feel so worthless.',
    'No one likes me.',
    "I can't do anything.",
    'I am so useless',
    'Nothing makes sense anymore'],
   'responses': ["It's only natural to feel this way. Tell me more.
What else is on your mind?",
    "Let's discuss further why you're feeling this way.",
    'I first want to let you know that you are not alone in your
feelings and there is always someone there to help . you can always
change your feelings and change your way of thinking by being open to
trying to change.',
    'i first want to let you know that you are not alone in your
feelings and there is always someone there to help . you can always
change your feelings and change your way of thinking by being open to
trying to change.']},
  {'tag': 'depressed',
   'patterns': ["I can't take it anymore",
    'I am so depressed',
    "I think i'm depressed.",
    'I have depression'],
   'responses': ["It helps to talk about what's happening. You're
going to be okay",
    'Talk to me. Tell me more. It helps if you open up yourself to
someone else.',
    'Sometimes when we are depressed, it is hard to care about
anything. It can be hard to do the simplest of things. Give yourself
time to heal.']},
```

```
  {'tag': 'happy',
   'patterns': ['I feel great today.',
    'I am happy.',
    'I feel happy.',
    "I'm good.",
    'cheerful',
    "I'm fine",
    'I feel ok'],
   'responses': ["That's geat to hear. I'm glad you're feeling this
way.",
    "Oh i see. That's great.",
    'Did something happen which made you feel this way?']},
  {'tag': 'casual',
   'patterns': ['Oh I see.',
    'ok',
    'okay',
    'nice',
    'Whatever',
    'K',
    'Fine',
    'yeah',
    'yes',
    'no',
    'not really'],
   'responses': ["Let's discuss further why you're feeling this way.",
    'How were you feeling last week?',
    "I'm listening. Please go on.",
    'Tell me more',
    'Can you elaborate on that?',
    'Come Come elucidate your thoughts']},
  {'tag': 'anxious',
   'patterns': ['I feel so anxious.', "I'm so anxious because of "],
   'responses': ["Don't be hard on yourself. What's the reason behind
this?",
    'Can you tell me more about this feeling?',
    'I understand that it can be scary. Tell me more about it.',
    "Don't let the little worries bring you down. What's the worse
that can happen?"]},
  {'tag': 'not-talking',
   'patterns': ["I don't want to talk about it.",
    'No just stay away.',
    "I can't bring myself to open up.",
    'Just shut up'],
   'responses': ["Talking about something really helps. If you're not
ready to open up then that's ok. Just know that i'm here for you,
whenever you need me.",
    "I want to help you. I really do. But in order for me to help you,
you're gonna have to talk to me.",
    "I'm here to listen to you and help you vent. So please talk to
```

```
me.",
    'You can talk to me without fear of judgement.']},
  {'tag': 'sleep',
   'patterns': ['I have insominia',
    'I am suffering from insomnia',
    "I can't sleep.",
    "I haven't slept for the last days.",
    "I can't seem to go to sleep.",
    "I haven't had proper sleep for the past few days."],
   'responses': ['What do you think is the reason behind this?',
    'That seem awful. What do you think is behind this?']},
  {'tag': 'scared',
   'patterns': ["I'm scared",
    'That sounds awful. What do i do?',
    "No i don't want to feel this way",
    'I am scared for myself'],
   'responses': ["It's only natural to feel this way. I'm here for
you.",
    "It'll all be okay. This feeling is only momentary.",
    "I understand how you feel. Don't put yourself down because of
it."]},
  {'tag': 'death',
   'patterns': ['My mom died',
    'My brother died',
    'My dad passed away',
    'My sister passed away',
    'Someone in my family died',
    'My friend passed away'],
   'responses': ["I'm sorry to hear that. If you want to talk about
it. I'm here.",
    'I am really sorry to hear that. I am here to help you with grief,
anxiety and anything else you may feel at this time.',
    "My condolences. I'm here if you need to talk."]},
  {'tag': 'understand',
   'patterns': ["You don't understand me.",
    "You're just some robot. How would you know?",
    "You can't possibly know what i'm going through",
    "You're useless",
    "You can't help me",
    'Nobody understands me.'],
   'responses': ["It sound like i'm not being very helpful right
now.",
    "I'm sorry to hear that. I'm doing my best to help",
    "I'm trying my best to help you. So please talk to me"]},
  {'tag': 'done',
   'patterns': ["That's all.",
    "I don't have anything more to say",
    'Nothing else',
    "That's all i have to say",
```

```
    'no, that would be all'],
   'responses': ['I heard you & noted it all. See you later.',
    "Oh okay we're done for today then. See you later",
    'I hope you have a great day. See you soon',
    "Okay we're done. Have a great day",
    'Okay I see. Enjoy the rest of your day then']},
  {'tag': 'suicide',
   'patterns': ['I want to kill myself',
    "I've thought about killing myself.",
    'I want to die',
    'I am going to kill myself',
    'I am going to commit suicide'],
   'responses': ["I'm very sorry to hear that but you have so much to
look forward to. Please seek help by contacting: 9152987821."]},
  {'tag': 'hate-you',
   'patterns': ['I hate you', "I don't like you", "I don't trust
you"],
   'responses': ["I'm sorry if i offended you in anyway. I'm only here
to help",
    'Forgive me if i did anything to offend you. I only want to
help']},
  {'tag': 'hate-me',
   'patterns': ['You hate me', 'I know you hate me', "You don't like
me"],
   'responses': ['Why do you think so?',
    "I'm sorry if i have exhibited any sort of behaviour to make you
think that."]},
  {'tag': 'default',
   'patterns': ['exams',
    'friends',
    'relationship',
    'boyfriend',
    'girlfriend',
    'family',
    'money',
    'financial problems'],
   'responses': ['Oh I see. Tell me more',
    'I see. What else?',
    'Tell me more about it.',
    "Oh okay. Why don't you tell me more about it?",
    "I'm listening. Tell me more."]},
  {'tag': 'jokes',
   'patterns': ['Tell me a joke', 'Tell me another joke'],
   'responses': ['mental health is not a joke.']},
  {'tag': 'repeat',
   'patterns': ['You already told me that',
    'You mentioned that already',
    'Why are you repeating yourself?'],
   'responses': ["Oh sorry I didn't realise that. I'll try not to
```

```
repeat myself again."]},
  {'tag': 'wrong',
   'patterns': ['What are you saying?',
    "That doesn't make sense",
    'Wrong response',
    'Wrong answer'],
   'responses': ["I'm very sorry. Let's try that again"]},
  {'tag': 'stupid',
   'patterns': ['Are you stupid?',
    "You're crazy",
    'You are dumb',
    'Are you dumb?'],
   'responses': ["I wish you wouldn't say such hurtful things. I'm
sorry if I wasn't useful"]},
  {'tag': 'location',
   'patterns': ['Where are you?',
    'Where do you live?',
    'What is your location?'],
   'responses': ['Duh I live in your computer',
    'Everywhere',
    'Somewhere in the universe']},
  {'tag': 'something-else',
   'patterns': ['I want to talk about something else',
    "Let's talk about something else.",
    'Can we not talk about this?',
    "I don't want to talk about this."],
   'responses': ['Okay sure. What do you want to talk about?',
    'Alright no problem. Is there something you want to talk about?',
    'Is there something else that you want to talk about?']},
  {'tag': 'friends',
   'patterns': ["I don't have any friends"],
   'responses': ["I'm sorry to hear that. Just know that I'm here for
you. Talking about it might help. Why do you think you don't have any
friends?"]},
  {'tag': 'ask',
   'patterns': ['Can I ask you something?'],
   'responses': ["Sure. I'll try my best to answer you",
    "Of course. Feel free to ask me anything. I'll do my best to
answer you"]},
  {'tag': 'problem',
   'patterns': ["Probably because my exams are approaching. I feel
stressed out because I don't think I've prepared well enough.",
    'probably because of my exams'],
   'responses': ['I see. Have you taken any approaches to not feel
this way?']},
  {'tag': 'no-approach',
   'patterns': ['I guess not. All I can think about are my exams.',
    'not really',
    'i guess not'],
```

```
   'responses': ["That's no problem. I can see why you'd be stressed
out about that. I can suggest you some tips to alleviate this issue.
Would you like to learn more about that?"]},
  {'tag': 'learn-more',
   'patterns': ['ok sure. i would like to learn more about it.',
    'yes, i would like to learn more about it.',
    'i would like to learn more about it.'],
   'responses': ["So first I would suggest you to give yourself a
break. Thinking more and more about the problem definitely does not
help in solving it. You'll just end up overwhelming yourself."]},
  {'tag': 'user-agree',
   'patterns': ["yeah you're right. i deserve a break.",
    "Yeah you're absolutely right about that"],
   'responses': ['Next, I would suggest you to practice meditation.
Meditation can produce a deep state of relaxation and a tranquil
mind.']},
  {'tag': 'meditation',
   'patterns': ['hmmm that sounds like it could be useful to me.',
    'That sounds useful.'],
   'responses': ['Focus all your attention on your breathing.
Concentrate on feeling and listening as you inhale and exhale through
your nostrils. Breathe deeply and slowly. When your attention wanders,
gently return your focus to your breathing.']},
  {'tag': 'user-meditation',
   'patterns': ['i did what you said and i feel alot better. thank you
very much.',
    'I feel better now'],
   'responses': ["Your welcome. Remember: Always focus on what's
within your control. When you find yourself worrying, take a minute to
examine the things you have control over. You can't prevent a storm
from coming but you can prepare for it. You can't control how someone
else behaves, but you can control how you react. Recognize that
sometimes, all you can control is your effort and your attitude. When
you put your energy into the things you can control, you'll be much
more effective."]},
  {'tag': 'pandora-useful',
   'patterns': ["thank you very much again. i'll continue practicing
meditation and focus on what i can control."],
   'responses': ["I'm glad you found this useful. Is there something
else I can help you with?"]},
  {'tag': 'user-advice',
   'patterns': ['I want some advice.',
    'I need some advice.',
    'I need advice on something'],
   'responses': ['Sure. What can I do to help?',
    'Okay what do you need advice on?']},
  {'tag': 'learn-mental-health',
   'patterns': ['I want to learn about mental health.',
    'I want to learn more about mental health.',
```

```
    "I'm interested in learning about mental health."],
   'responses': ["Oh that's really great. I'd be willing to answer
anything that I know about it."]},
  {'tag': 'mental-health-fact',
   'patterns': ['Tell me a fact about mental health',
    'Tell me another fact about mental health'],
   'responses': ['According to a UNICEF report, One in seven Indians
between 15-24 years of age feels depressed',
    '1 in 5 young people (age 13-18) has or will develop a mental
illness in their lifetime.',
    'Depression is the leading cause of disability worldwide.']},
  {'tag': 'mental-health-definition',
   'patterns': ['What is mental health?', 'Define Mental Health'],
   'responses': ['Mental health is a state of well-being in which the
individual realizes his or her own abilities, can cope with the normal
stresses of life, can work productively and fruitfully, and is able to
make a contribution to his or her community',
    'Mental health includes our emotional, psychological, and social
well-being. It affects how we think, feel, and act. It also helps
determine how we handle stress, relate to others, and make
choices.']},
  {'tag': 'importance',
   'patterns': ['Why is mental health important?',
    'What is the importance of mental health?'],
   'responses': ['Maintaining mental health is crucial to stabilizing
constructive behaviors, emotions, and thoughts. Focusing on mental
health care can increase productivity, enhance our self-image, and
improve relationships.']},
  {'tag': 'depression-definition',
   'patterns': ['What is Depression?', 'Define Depression'],
   'responses': ['A mental health disorder characterised by
persistently depressed mood or loss of interest in activities, causing
significant impairment in daily life.']},
  {'tag': 'depression-diagnosis',
   'patterns': ['How do i know if i have Depression?',
    'Am i depressed?',
    'Am i suffering from depression?',
    'Am i mentally ill?'],
   'responses': ['For a diagnosis of depression, a person needs to
have experienced low mood or loss of interest or pleasure in life for
at least 2 weeks. Also, they will have experienced the following
symptoms: feelings of sadness, hopelessness, or irritability nearly
every day.']},
  {'tag': 'therapist',
   'patterns': ['What is a therapist?', 'What does a therapist do?'],
   'responses': ['A therapist is a broad designation that refers to
professionals who are trained to provide treatment and rehabilitation.
The term is often applied to psychologists, but it can include others
who provide a variety of services, including social workers,
```

```
counselors, life coaches, and many others. ']},
  {'tag': 'therapy',
   'patterns': ['What is therapy?',
    'Do i need therapy?',
    'Who is therapy for?'],
   'responses': ['Therapy is a form of treatment that aims to help
resolve mental or emotional issues.',
    'Therapy is a form of treatment that aims to help resolve mental
or emotional issues. It is helpful for those with mental health
conditions or even everyday life challenges.']},
  {'tag': 'mental-illness',
   'patterns': ['What does it mean to have a mental illness?'],
   'responses': ["Mental illnesses are health conditions that disrupt
a person's thoughts, emotions, relationships, and daily functioning.
They are associated with distress and diminished capacity to engage in
the ordinary activities of daily life. Mental illnesses fall along a
continuum of severity: some are fairly mild and only interfere with
some aspects of life, such as certain phobias. On the other end of the
spectrum lie serious mental illnesses, which result in major
functional impairment and interference with daily life. These include
such disorders as major depression, schizophrenia, and bipolar
disorder, and may require that the person receives care in a hospital.
It is important to know that mental illnesses are medical conditions
that have nothing to do with a person's character, intelligence, or
willpower. Just as diabetes is a disorder of the pancreas, mental
illness is a medical condition due to the brain's biology. Similarly
to how one would treat diabetes with medication and insulin, mental
illness is treatable with a combination of medication and social
support. These treatments are highly effective, with 70-90 percent of
individuals receiving treatment experiencing a reduction in symptoms
and an improved quality of life. With the proper treatment, it is very
possible for a person with mental illness to be independent and
successful."]},
  {'tag': 'mental-illness-affect',
   'patterns': ['Who does mental illness affect?'],
   'responses': ['It is estimated that mental illness affects 1 in 5
adults in America, and that 1 in 24 adults have a serious mental
illness. Mental illness does not discriminate; it can affect anyone,
regardless of gender, age, income, social status, ethnicity, religion,
sexual orientation, or background. Although mental illness can affect
anyone, certain conditions may be more common in different
populations. For instance, eating disorders tend to occur more often
in females, while disorders such as attention deficit/hyperactivity
disorder is more prevalent in children. Additionally, all ages are
susceptible, but the young and the old are especially vulnerable.
Mental illnesses usually strike individuals in the prime of their
lives, with 75 percent of mental health conditions developing by the
age of 24. This makes identification and treatment of mental disorders
particularly difficult, because the normal personality and behavioral
changes of adolescence may mask symptoms of a mental health condition.
```

Parents and caretakers should be aware of this fact, and take notice of changes in their childÃ¢â‚¬â„¢s mood, personality, personal habits, and social withdrawal. When these occur in children under 18, they are referred to as serious emotional disturbances (SEDs).']},
  {'tag': 'mental-illness-cause',
   'patterns': ['What causes mental illness?'],
   'responses': ["It is estimated that mental illness affects 1 in 5 adults in America, and that 1 in 24 adults have a serious mental illness. Mental illness does not discriminate; it can affect anyone, regardless of gender, age, income, social status, ethnicity, religion, sexual orientation, or background. Although mental illness can affect anyone, certain conditions may be more common in different populations. For instance, eating disorders tend to occur more often in females, while disorders such as attention deficit/hyperactivity disorder is more prevalent in children. Additionally, all ages are susceptible, but the young and the old are especially vulnerable. Mental illnesses usually strike individuals in the prime of their lives, with 75 percent of mental health conditions developing by the age of 24. This makes identification and treatment of mental disorders particularly difficult, because the normal personality and behavioral changes of adolescence may mask symptoms of a mental health condition. Parents and caretakers should be aware of this fact, and take notice of changes in their child's mood, personality, personal habits, and social withdrawal. When these occur in children under 18, they are referred to as serious emotional disturbances (SEDs)."]},
  {'tag': 'mental-illness-symptoms',
   'patterns': ['What are some of the warning signs of mental illness?'],
   'responses': ['Symptoms of mental health disorders vary depending on the type and severity of the condition. The following is a list of general symptoms that may suggest a mental health disorder, particularly when multiple symptoms are expressed at once. \n In adults:\n Confused thinking\n Long-lasting sadness or irritability\n Extreme highs and lows in mood\n Excessive fear, worrying, or anxiety\n Social withdrawal\n Dramatic changes in eating or sleeping habits\n Strong feelings of anger\n Delusions or hallucinations (seeing or hearing things that are not really there)\n Increasing inability to cope with daily problems and activities\n Thoughts of suicide\n Denial of obvious problems\n Many unexplained physical problems\n Abuse of drugs and/or alcohol\n \nIn older children and pre-teens:\n Abuse of drugs and/or alcohol\n Inability to cope with daily problems and activities\n Changes in sleeping and/or eating habits\n Excessive complaints of physical problems\n Defying authority, skipping school, stealing, or damaging property\n Intense fear of gaining weight\n Long-lasting negative mood, often along with poor appetite and thoughts of death\n Frequent outbursts of anger\n \nIn younger children:\n Changes in school performance\n Poor grades despite strong efforts\n Excessive worrying or anxiety\n Hyperactivity\n Persistent nightmares\n Persistent disobedience and/or aggressive behavior\n

```
Frequent temper tantrums']},
  {'tag': 'mental-illness-recovery',
   'patterns': ['Can people with mental illness recover?'],
   'responses': ['When healing from mental illness, early
identification and treatment are of vital importance. Based on the
nature of the illness, there are a range of effective treatments
available. For any type of treatment, it is essential that the person
affected is proactive and fully engaged in their own recovery process.
Many people with mental illnesses who are diagnosed and treated
respond well, although some might experience a return of symptoms.
Even in such cases, with careful monitoring and management of the
disorder, it is still quite possible to live a fulfilled and
productive life.']},
  {'tag': 'mental-illness-advice',
   'patterns': ['What should I do if I know someone who appears to
have the symptoms of a mental disorder?'],
   'responses': ["Although Pandora cannot substitute for professional
advice, we encourage those with symptoms to talk to their friends and
family members and seek the counsel of a mental health professional.
The sooner the mental health condition is identified and treated, the
sooner they can get on the path to recovery. If you know someone who
is having problems, don't assume that the issue will resolve itself.
Let them know that you care about them, and that there are treatment
options available that will help them heal. Speak with a mental health
professional or counselor if you think your friend or family member is
experiencing the symptoms of a mental health condition. If the
affected loved one knows that you support them, they will be more
likely to seek out help."]},
  {'tag': 'mental-health-professionals',
   'patterns': ['How can I find a mental health professional for
myself or my child?'],
   'responses': ['Feeling comfortable with the professional you or
your child is working with is critical to the success of the
treatment. Finding the professional who best fits your needs may
require research. Start by searching for providers in your area.']},
  {'tag': 'treatment',
   'patterns': ['What treatment options are available?'],
   'responses': ['Just as there are different types of medications for
physical illness, different treatment options are available for
individuals with mental illness. Treatment works differently for
different people. It is important to find what works best for you or
your child.']},
  {'tag': 'treatment-facts',
   'patterns': ['If I become involved in treatment, what do I need to
know?'],
   'responses': ["Since beginning treatment is a big step for
individuals and families, it can be very overwhelming. It is important
to be as involved and engaged in the treatment process as possible.
Some questions you will need to have answered include:\n What is known
```

```
about the cause of this particular illness?\n Are there other
diagnoses where these symptoms are common?\n Do you normally include a
physical or neurological examination?\n Are there any additional tests
or exams that you would recommend at this point?\n Would you advise an
independent opinion from another psychiatrist at this point?\n What
program of treatment is the most helpful with this diagnosis?\n Will
this program involve services by other specialists? If so, who will be
responsible for coordinating these services?\n What do you see as the
family's role in this program of treatment?\n How much access will the
family have to the individuals who are providing the treatment?\n What
medications are generally used with this diagnosis?\n How much
experience do you have in treating individuals with this illness?\n
What can I do to help you in the treatment?"]},
  {'tag': 'professionals-type',
   'patterns': ['What is the difference between mental health
professionals?'],
   'responses': ['There are many types of mental health professionals.
The variety of providers and their services may be confusing. Each
have various levels of education, training, and may have different
areas of expertise. Finding the professional who best fits your needs
may require some research.']},
  {'tag': 'finding-professionals',
   'patterns': ['How can I find a mental health professional right for
my child or myself?'],
   'responses': ['Feeling comfortable with the professional you or
your child is working with is critical to the success of your
treatment. Finding the professional who best fits your needs may
require some research.']},
  {'tag': 'other-options',
   'patterns': ['Where else can I get help?'],
   'responses': ['Where you go for help will depend on the nature of
the problem and/or symptoms and what best fits you. Often, the best
place to start is by talking with someone you trust about your
concerns, such as a family member, friend, clergy, healthcare
provider, or other professionals. Having this social support is
essential in healing from mental illness, and you will be able to ask
them for referrals or recommendations for trusted mental health
practitioners. Search for mental health resources in your area.
Secondly, there are people and places throughout the country that
provide services to talk, to listen, and to help you on your journey
to recovery. Thirdly, many people find peer support a helpful tool
that can aid in their recovery. There are a variety of organizations
that offer support groups for consumers, their family members, and
friends. Some support groups are peer led while others may be led by a
mental health professional.']},
  {'tag': 'medication-facts',
   'patterns': ['What should I know before starting a new
medication?'],
   'responses': ['The best source of information regarding medications
```

is the physician prescribing them. He or she should be able to answer questions such as: \n1. What is the medication supposed to do? \n2. When should it begin to take effect, and how will I know when it is effective? \n3. How is the medication taken and for how long? What food, drinks, other medicines, and activities should be avoided while taking this medication? \n4. What are the side effects and what should be done if they occur? \n5. What do I do if a dose is missed? \n6. Is there any written information available about this medication? \n7. Are there other medications that might be appropriate? \n8. If so, why do you prefer the one you have chosen? \n9. How do you monitor medications and what symptoms indicate that they should be raised, lowered, or changed? \n10. All medications should be taken as directed. Most medications for mental illnesses do not work when taken irregularly, and extra doses can cause severe, sometimes dangerous side effects. Many psychiatric medications begin to have a beneficial effect only after they have been taken for several weeks.']},
  {'tag': 'therapy-places',
   'patterns': ['Where can I go to find therapy?'],
   'responses': ['Different kinds of therapy are more effective based on the nature of the mental health condition and/or symptoms and the person who has them (for example, children will benefit from a therapist who specializes in childrenâ€™s mental health). However, there are several different types of treatment and therapy that can help.']},
  {'tag': 'mental-health-knowledge',
   'patterns': ['Where can I learn about types of mental health treatment?'],
   'responses': ['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.']},
  {'tag': 'mental-health-advisors',
   'patterns': ['What are the different types of mental health professionals?'],
   'responses': ['There are many types of mental health professionals. Finding the right one for you may require some research.']},
  {'tag': 'support-group',
   'patterns': ['Where can I go to find a support group?'],
   'responses': ['Many people find peer support a helpful tool that can aid in their recovery. There are a variety of organizations that offer support groups for consumers, their family members and friends. Some support groups are peer-led, while others may be led by a mental health professional.']},
  {'tag': 'mental-health-precautions',
   'patterns': ['Can you prevent mental health problems?'],
   'responses': ['We can all suffer from mental health challenges, but developing our wellbeing, resilience, and seeking help early can help

```
prevent challenges becoming serious.']},
  {'tag': 'cure',
   'patterns': ['Are there cures for mental health problems?',
    'is there any cure for mental health problems?'],
   'responses': ['It is often more realistic and helpful to find out
what helps with the issues you face. Talking, counselling, medication,
friendships, exercise, good sleep and nutrition, and meaningful
occupation can all help.']},
  {'tag': 'mental-health-problems',
   'patterns': ['What causes mental health problems?'],
   'responses': ['Challenges or problems with your mental health can
arise from psychological, biological, and social, issues, as well as
life events.']},
  {'tag': 'mental-health-worries',
   'patterns': ["What do I do if I'm worried about my mental
health?"],
   'responses': ['The most important thing is to talk to someone you
trust. This might be a friend, colleague, family member, or GP. In
addition to talking to someone, it may be useful to find out more
information about what you are experiencing. These things may help to
get some perspective on what you are experiencing, and be the start of
getting help.']},
  {'tag': 'self-awareness',
   'patterns': ["How do I know if I'm unwell?"],
   'responses': ['If your beliefs , thoughts , feelings or behaviours
have a significant impact on your ability to function in what might be
considered a normal or ordinary way, it would be important to seek
help.']},
  {'tag': 'loneliness',
   'patterns': ['How can I maintain social connections? What if I feel
lonely?'],
   'responses': ["A lot of people are alone right now, but we don't
have to be lonely. We're all in this together. Think about the
different ways to connect that are most meaningful for you. For
example, you might prefer a video chat over a phone call, or you might
prefer to text throughout the day rather than one set time for a video
call. Then, work with your social networks to make a plan. You might
video chat with your close friends in the evening and phone a family
member once a week. Remember to be mindful of people who may not be
online. Check in by phone and ask how you can help. The quality of
your social connections matter. Mindlessly scrolling through social
media and liking a few posts usually doesn't build strong social
connections. Make sure you focus on strategies that actually make you
feel included and connected. If your current strategies don't help you
feel connected, problem-solve to see if you can find a solution.
Everyone feels lonely at times. Maybe you recently moved to a new
city, are changing your circle of friends, lost someone important in
your life, or lost your job and also lost important social connections
with coworkers. Other people may have physical connections to others
```

```
but may feel like their emotional or social needs aren't met. Measures like social distancing or self-isolation can make loneliness feel worse no matter why you feel lonely now. Reach out to the connections you do have. Suggest ways to keep in touch and see if you can set a regular time to connect. People may hesitate to reach out for a lot of different reasons, so don't be afraid to be the one who asks. Look for local community support groups and mutual aid groups on social media. This pandemic is bringing everyone together, so look for opportunities to make new connections. These groups are a great way to share your skills and abilities or seek help and support. Look for specialized support groups. Support groups are moving online, and there are a lot of different support lines to call if you need to talk to someone."]},
  {'tag': 'anxiety-stress',
   'patterns': ["What's the difference between anxiety and stress?"],
   'responses': ["Stress and anxiety are often used interchangeably, and there is overlap between stress and anxiety. Stress is related to the same fight, flight, or freeze response as anxiety, and the physical sensations of anxiety and stress may be very similar. The cause of stress and anxiety are usually different, however. Stress focuses on mainly external pressures on us that we're finding hard to cope with. When we are stressed, we usually know what we're stressed about, and the symptoms of stress typically disappear after the stressful situation is over. Anxiety, on the other hand, isn't always as easy to figure out. Anxiety focuses on worries or fears about things that could threaten us, as well as anxiety about the anxiety itself. Stress and anxiety are both part of being human, but both can be problems if they last for a long time or have an impact on our well-being or daily life."]},
  {'tag': 'sadness-depression',
   'patterns': ["What's the difference between sadness and depression?",
    'difference between sadness and depression'],
   'responses': ["Sadness is a normal reaction to a loss, disappointment, problems, or other difficult situations. Feeling sad from time to time is just another part of being human. In these cases, feelings of sadness go away quickly and you can go about your daily life. Other ways to talk about sadness might be feeling low, feeling down, or feeling blue.A person may say they are feeling depressed, but if it goes away on its own and doesn't impact life in a big way, it probably isn't the illness of depression. Depression is a mental illness that affects your mood, the way you understand yourself, and the way you understand and relate to things around you. It can also go by different names, such as clinical depression, major depressive disorder, or major depression. Depression can come up for no reason, and it lasts for a long time. It's much more than sadness or low mood. People who experience depression may feel worthless or hopeless. They may feel unreasonable guilty. Some people may experience depression as anger or irritability. It may be hard to concentrate or make decisions. Most people lose interest in things that they used to enjoy
```

```
and may isolate themselves from others. There are also physical signs of depression, such as problems with sleep, appetite and energy and unexplainable aches or pains. Some may experience difficult thoughts about death or ending their life (suicide). Depression lasts longer than two weeks, doesn't usually go away on its own, and impacts your life. It's a real illness, and it is very treatable. It's important to seek help if you're concerned about depression."]}]}

# Initialize WordNetLemmatizer
lemmatizer = WordNetLemmatizer()

# Initialize lists to store words, classes, and documents
word_list = []
class_list = []
document_list = []
ignore_words = ['?']

# Iterate through intents and patterns
for intent_data in intents['intents']:
    for pattern in intent_data['patterns']:
        # Tokenize each word in the sentence
        tokenized_words = nltk.word_tokenize(pattern)
        # Add words to the word list
        word_list.extend(tokenized_words)
        # Add documents to the corpus
        document_list.append((tokenized_words, intent_data['tag']))
        # Add classes to the class list
        if intent_data['tag'] not in class_list:
            class_list.append(intent_data['tag'])

# Lemmatize words and filter out ignored words
word_list = [lemmatizer.lemmatize(tokenized_words.lower()) for
tokenized_words in word_list if tokenized_words not in ignore_words]
# Remove duplicates and sort words
word_list = sorted(list(set(word_list)))
class_list = sorted(list(set(class_list)))

# Print summary information
print(len(document_list), "documents")
print(len(class_list), "classes", class_list)
print(len(word_list), "unique lemmatized words", word_list)

232 documents
80 classes ['about', 'afternoon', 'anxiety-stress', 'anxious', 'ask',
'casual', 'creation', 'cure', 'death', 'default', 'depressed',
'depression-definition', 'depression-diagnosis', 'done', 'evening',
'finding-professionals', 'friends', 'goodbye', 'greeting', 'happy',
'hate-me', 'hate-you', 'help', 'importance', 'jokes', 'learn-mental-
health', 'learn-more', 'location', 'loneliness', 'medication-facts',
'meditation', 'mental-health-advisors', 'mental-health-definition',
```

```
'mental-health-fact', 'mental-health-knowledge', 'mental-health-
precautions', 'mental-health-problems', 'mental-health-professionals',
'mental-health-worries', 'mental-illness', 'mental-illness-advice',
'mental-illness-affect', 'mental-illness-cause', 'mental-illness-
recovery', 'mental-illness-symptoms', 'morning', 'name', 'neutral-
response', 'night', 'no-approach', 'no-response', 'not-talking',
'other-options', 'pandora-useful', 'problem', 'professionals-type',
'repeat', 'sad', 'sadness-depression', 'scared', 'self-awareness',
'skill', 'sleep', 'something-else', 'stressed', 'stupid', 'suicide',
'support-group', 'thanks', 'therapist', 'therapy', 'therapy-places',
'treatment', 'treatment-facts', 'understand', 'user-advice', 'user-
agree', 'user-meditation', 'worthless', 'wrong']
297 unique lemmatized words ["'ll", "'m", "'re", "'s", "'ve", ',',
'.', 'a', 'about', 'absolutely', 'advice', 'affect', 'afternoon',
'again', 'all', 'alot', 'already', 'am', 'and', 'another', 'answer',
'anxiety', 'anxious', 'any', 'anymore', 'anyone', 'anything',
'appears', 'approaching', 'are', 'ask', 'au', 'available', 'away',
'awful', 'be', 'because', 'become', 'before', 'better', 'between',
'bonjour', 'boyfriend', 'break', 'bring', 'brother', 'burned', 'by',
'bye', 'ca', 'call', 'can', 'cause', 'cheerful', 'child', 'commit',
'connection', 'continue', 'control', 'could', 'crazy', 'created',
'cure', 'dad', 'day', 'define', 'depressed', 'depression', 'deserve',
'did', 'die', 'died', 'difference', 'different', 'disorder', 'do',
'doe', 'down', 'dumb', 'else', 'empty', 'enough', 'evening', 'exam',
'fact', 'family', 'fare', 'feel', 'feeling', 'few', 'financial',
'find', 'fine', 'focus', 'for', 'friend', 'from', 'get', 'girlfriend',
'give', 'go', 'going', 'good', 'goodbye', 'great', 'group', 'guess',
'guten', 'had', 'hand', 'happy', 'hate', 'have', 'health', 'hello',
'help', 'helpful', 'hey', 'hi', 'hmmm', 'hola', 'how', 'howdy', 'i',
'if', 'ill', 'illness', 'importance', 'important', 'in', 'insominia',
'insomnia', 'interested', 'involved', 'is', 'it', 'joke', 'just', 'k',
'kill', 'killing', 'know', 'konnichiwa', 'last', 'later', 'learn',
'learning', 'let', 'like', 'live', 'location', 'lonely', 'made',
'maintain', 'make', 'me', 'mean', 'medication', 'meditation',
'mental', 'mentally', 'mentioned', 'mom', 'money', 'more', 'morning',
'much', 'my', 'myself', "n't", 'name', 'need', 'new', 'nice', 'night',
'no', 'nobody', 'not', 'nothing', 'now', 'of', 'oh', 'ok', 'okay',
'ola', 'on', 'one', 'open', 'option', 'or', 'out', 'passed', 'past',
'people', 'please', 'possibly', 'practicing', 'prepared', 'prevent',
'probably', 'problem', 'professional', 'proper', 'really', 'recover',
'relationship', 'repeating', 'response', 'revoir', 'right', 'robot',
'sad', 'sadness', 'said', 'say', 'saying', 'sayonara', 'scared',
'see', 'seem', 'sense', 'should', 'shut', 'sign', 'sister', 'sleep',
'slept', 'so', 'social', 'some', 'someone', 'something', 'sound',
'starting', 'stay', 'still', 'stress', 'stressed', 'stuck', 'stupid',
'suffering', 'suicide', 'support', 'sure', 'symptom', 'tag', 'take',
'talk', 'tell', 'than', 'thank', 'thanks', 'that', 'the', 'thee',
'then', 'therapist', 'therapy', 'there', 'think', 'this', 'thought',
'through', 'to', 'today', 'told', 'treatment', 'trust', 'type',
```

```
'understand', 'understands', 'unwell', 'up', 'useful', 'useless',
'very', 'want', 'warning', 'way', 'we', 'well', 'were', 'what',
'whatever', 'where', 'who', 'why', 'with', 'worried', 'worthless',
'would', 'wrong', 'yeah', 'yes', 'you', 'your', 'yourself']

# Initialize empty lists for training, output, and output_empty
training_list = []
output_list = []
output_empty_list = [0] * len(class_list)

# Create bag of words for each sentence in the training set
for document in document_list:
    # Initialize bag of words
    bag_of_words = []
    # Tokenize words in the pattern
    pattern_words = document[0]
    # Lemmatize each word
    pattern_words = [lemmatizer.lemmatize(word.lower()) for word in
pattern_words]
    # Create bag of words array
    for word in word_list:
        bag_of_words.append(1) if word in pattern_words else
bag_of_words.append(0)

    # Create output row: '0' for each tag and '1' for the current tag
    output_row = list(output_empty_list)
    output_row[class_list.index(document[1])] = 1

    # Append bag of words and output row to the training list
    training_list.append([bag_of_words, output_row])

# Shuffle the training list
random.shuffle(training_list)

# Function to perform synonym replacement augmentation
def synonym_replacement(tokens, limit):
    augmented_sentences = []
    for i in range(len(tokens)):
        synonyms = []
        for syn in wordnet.synsets(tokens[i]):
            for lemma in syn.lemmas():
                synonyms.append(lemma.name())
        if len(synonyms) > 0:
            num_augmentations = min(limit, len(synonyms))
            sampled_synonyms = random.sample(synonyms,
num_augmentations)
            for synonym in sampled_synonyms:
                augmented_tokens = tokens[:i] + [synonym] + tokens[i +
1:]
                augmented_sentences.append(' '.join(augmented_tokens))
```

```
    return augmented_sentences

# Augment the training data using synonym replacement
augmented_data_list = []
limit_per_tag = 100

for index, document in enumerate(training_list):
    bag_of_words, output_row = document
    tokens = [word_list[j] for j in range(len(word_list)) if
bag_of_words[j] == 1]
    augmented_sentences = synonym_replacement(tokens, limit_per_tag)
    for augmented_sentence in augmented_sentences:
        augmented_bag_of_words = [1 if augmented_sentence.find(word)
>= 0 else 0 for word in word_list]
        augmented_data_list.append([augmented_bag_of_words,
output_row])

augmented_data_list = augmented_data_list[:len(training_list)]

# Combine the datasets
combined_data_list = training_list + augmented_data_list
random.shuffle(combined_data_list)

from sklearn.model_selection import train_test_split
def group_data_by_tags(data):
    grouped_data = {}
    for item in data:
        tag = tuple(item[1])
        if tag not in grouped_data:
            grouped_data[tag] = []
        grouped_data[tag].append(item)
    return grouped_data.values()

grouped_data = group_data_by_tags(combined_data_list)

from sklearn.model_selection import train_test_split

# Lists to store training and testing data
training_data_list = []
testing_data_list = []

# Split each tag's data into training and testing sets
for tag_data in grouped_data:
    # Check if there are sufficient samples to split
    if len(tag_data) > 1:
        train_data, test_data = train_test_split(tag_data,
test_size=0.1, random_state=21)
        training_data_list.extend(train_data)
        testing_data_list.extend(test_data)
    else:
```

```
        # If there's only one sample, add it to the training set
        training_data_list.extend(tag_data)

# Shuffle training and testing data
random.shuffle(training_data_list)
random.shuffle(testing_data_list)

# Convert training and testing data back to np.array
train_x = np.array([d[0] for d in training_data_list])
train_y = np.array([d[1] for d in training_data_list])
test_x = np.array([d[0] for d in testing_data_list])
test_y = np.array([d[1] for d in testing_data_list])

import torch
import torch.nn as nn
import torch.optim as optim
from torch.utils.data import Dataset, DataLoader
import numpy as np

class ComplexNeuralNetwork(nn.Module):
    def __init__(self, input_size, hidden_size1, hidden_size2,
output_size):
        super(ComplexNeuralNetwork, self).__init__()
        self.fc1 = nn.Linear(input_size, hidden_size1)
        self.relu1 = nn.ReLU()
        self.bn1 = nn.BatchNorm1d(hidden_size1)
        self.dropout1 = nn.Dropout(0.5)

        self.fc2 = nn.Linear(hidden_size1, hidden_size2)
        self.relu2 = nn.ReLU()
        self.bn2 = nn.BatchNorm1d(hidden_size2)
        self.dropout2 = nn.Dropout(0.5)

        self.fc3 = nn.Linear(hidden_size2, output_size)
        self.softmax = nn.Softmax(dim=1)

    def forward(self, x):
        x = self.fc1(x)
        x = self.relu1(x)
        x = self.bn1(x)
        x = self.dropout1(x)

        x = self.fc2(x)
        x = self.relu2(x)
        x = self.bn2(x)
        x = self.dropout2(x)

        x = self.fc3(x)
        output = self.softmax(x)
        return output
```

```
class CustomDataset(Dataset):
    def __init__(self, x, y):
        self.x = x
        self.y = y

    def __len__(self):
        return len(self.x)

    def __getitem__(self, idx):
        return self.x[idx], self.y[idx]

def compute_accuracy(predictions, targets):
    predicted_labels = torch.argmax(predictions, dim=1)
    true_labels = torch.argmax(targets, dim=1)
    correct = (predicted_labels == true_labels).sum().item()
    total = targets.size(0)
    return correct / total

def evaluate_model(model, data_loader, criterion):
    model.eval()
    total_loss = 0.0
    total_accuracy = 0.0
    num_batches = len(data_loader)

    with torch.no_grad():
        for inputs, targets in data_loader:
            outputs = model(inputs)
            loss = criterion(outputs, targets)
            total_loss += loss.item() * inputs.size(0)
            total_accuracy += compute_accuracy(outputs, targets) *
inputs.size(0)

    average_loss = total_loss / len(data_loader.dataset)
    average_accuracy = total_accuracy / len(data_loader.dataset)
    return average_loss, average_accuracy

# DataLoader for training and testing data
train_x = torch.tensor(train_x).float()
train_y = torch.tensor(train_y).float()
test_x = torch.tensor(test_x).float()
test_y = torch.tensor(test_y).float()

batch_size = 8
train_dataset = CustomDataset(train_x, train_y)
test_dataset = CustomDataset(test_x, test_y)
train_loader = DataLoader(train_dataset, batch_size=batch_size,
shuffle=True)
test_loader = DataLoader(test_dataset, batch_size=batch_size,
shuffle=False)
```

```
# Model, loss function, and optimizer
input_size = len(train_x[0])
hidden_size1 = 50
hidden_size2 = 30
output_size = len(train_y[0])
model = ComplexNeuralNetwork(input_size, hidden_size1, hidden_size2,
output_size)
criterion = nn.BCELoss()
optimizer = optim.Adam(model.parameters(), lr=0.001)  # Adjust
learning rate

# Training and evaluation
num_epochs = 250
best_accuracy = 0.0  # Track best accuracy
for epoch in range(num_epochs):
    # Training
    model.train()
    running_loss = 0.0
    running_acc = 0.0
    for inputs, targets in train_loader:
        optimizer.zero_grad()
        outputs = model(inputs)
        loss = criterion(outputs, targets)
        loss.backward()
        optimizer.step()
        running_loss += loss.item() * inputs.size(0)
        running_acc += compute_accuracy(outputs, targets) *
inputs.size(0)

    epoch_loss = running_loss / len(train_loader.dataset)
    epoch_acc = running_acc / len(train_loader.dataset)

    print(f"Epoch [{epoch+1}/{num_epochs}], Train Loss:
{epoch_loss:.4f}, Train Acc: {epoch_acc:.4f}")

    # Evaluation
    test_loss, test_accuracy = evaluate_model(model, test_loader,
criterion)
    print(f"Epoch [{epoch+1}/{num_epochs}], Test Loss:
{test_loss:.4f}, Test Acc: {test_accuracy:.4f}")

    # Save the model with the best accuracy
    if test_accuracy > best_accuracy:
        torch.save(model.state_dict(), 'best_model.pth')
        best_accuracy = test_accuracy

# Print best accuracy achieved
print(f"Best Test Accuracy: {best_accuracy:.4f}")
```

```
Epoch [1/250], Train Loss: 0.0675, Train Acc: 0.0691
Epoch [1/250], Test Loss: 0.0641, Test Acc: 0.1644
Epoch [2/250], Train Loss: 0.0586, Train Acc: 0.2506
Epoch [2/250], Test Loss: 0.0565, Test Acc: 0.3699
Epoch [3/250], Train Loss: 0.0526, Train Acc: 0.3836
Epoch [3/250], Test Loss: 0.0527, Test Acc: 0.3836
Epoch [4/250], Train Loss: 0.0470, Train Acc: 0.4450
Epoch [4/250], Test Loss: 0.0499, Test Acc: 0.3973
Epoch [5/250], Train Loss: 0.0447, Train Acc: 0.4680
Epoch [5/250], Test Loss: 0.0475, Test Acc: 0.3836
Epoch [6/250], Train Loss: 0.0414, Train Acc: 0.5064
Epoch [6/250], Test Loss: 0.0463, Test Acc: 0.3836
Epoch [7/250], Train Loss: 0.0389, Train Acc: 0.5141
Epoch [7/250], Test Loss: 0.0449, Test Acc: 0.3699
Epoch [8/250], Train Loss: 0.0383, Train Acc: 0.5166
Epoch [8/250], Test Loss: 0.0435, Test Acc: 0.3836
Epoch [9/250], Train Loss: 0.0363, Train Acc: 0.5192
Epoch [9/250], Test Loss: 0.0427, Test Acc: 0.3973
Epoch [10/250], Train Loss: 0.0355, Train Acc: 0.5320
Epoch [10/250], Test Loss: 0.0416, Test Acc: 0.3699
Epoch [11/250], Train Loss: 0.0347, Train Acc: 0.5396
Epoch [11/250], Test Loss: 0.0411, Test Acc: 0.4110
Epoch [12/250], Train Loss: 0.0341, Train Acc: 0.5473
Epoch [12/250], Test Loss: 0.0403, Test Acc: 0.4110
Epoch [13/250], Train Loss: 0.0333, Train Acc: 0.5601
Epoch [13/250], Test Loss: 0.0396, Test Acc: 0.3973
Epoch [14/250], Train Loss: 0.0330, Train Acc: 0.5780
Epoch [14/250], Test Loss: 0.0392, Test Acc: 0.4384
Epoch [15/250], Train Loss: 0.0323, Train Acc: 0.5729
Epoch [15/250], Test Loss: 0.0385, Test Acc: 0.4384
Epoch [16/250], Train Loss: 0.0325, Train Acc: 0.5550
Epoch [16/250], Test Loss: 0.0382, Test Acc: 0.4521
Epoch [17/250], Train Loss: 0.0313, Train Acc: 0.5678
Epoch [17/250], Test Loss: 0.0381, Test Acc: 0.4658
Epoch [18/250], Train Loss: 0.0313, Train Acc: 0.5780
Epoch [18/250], Test Loss: 0.0371, Test Acc: 0.4795
Epoch [19/250], Train Loss: 0.0287, Train Acc: 0.6138
Epoch [19/250], Test Loss: 0.0377, Test Acc: 0.4658
Epoch [20/250], Train Loss: 0.0284, Train Acc: 0.6036
Epoch [20/250], Test Loss: 0.0369, Test Acc: 0.4795
Epoch [21/250], Train Loss: 0.0298, Train Acc: 0.5934
Epoch [21/250], Test Loss: 0.0362, Test Acc: 0.4932
Epoch [22/250], Train Loss: 0.0295, Train Acc: 0.5985
Epoch [22/250], Test Loss: 0.0356, Test Acc: 0.4658
Epoch [23/250], Train Loss: 0.0297, Train Acc: 0.5780
Epoch [23/250], Test Loss: 0.0355, Test Acc: 0.4658
Epoch [24/250], Train Loss: 0.0289, Train Acc: 0.6138
Epoch [24/250], Test Loss: 0.0353, Test Acc: 0.4932
Epoch [25/250], Train Loss: 0.0280, Train Acc: 0.6061
Epoch [25/250], Test Loss: 0.0351, Test Acc: 0.4795
```

```
Epoch [26/250], Train Loss: 0.0267, Train Acc: 0.6292
Epoch [26/250], Test Loss: 0.0349, Test Acc: 0.5068
Epoch [27/250], Train Loss: 0.0264, Train Acc: 0.6138
Epoch [27/250], Test Loss: 0.0350, Test Acc: 0.5068
Epoch [28/250], Train Loss: 0.0258, Train Acc: 0.6343
Epoch [28/250], Test Loss: 0.0348, Test Acc: 0.4932
Epoch [29/250], Train Loss: 0.0273, Train Acc: 0.6215
Epoch [29/250], Test Loss: 0.0342, Test Acc: 0.5068
Epoch [30/250], Train Loss: 0.0262, Train Acc: 0.6240
Epoch [30/250], Test Loss: 0.0339, Test Acc: 0.5068
Epoch [31/250], Train Loss: 0.0269, Train Acc: 0.6138
Epoch [31/250], Test Loss: 0.0328, Test Acc: 0.5068
Epoch [32/250], Train Loss: 0.0264, Train Acc: 0.6087
Epoch [32/250], Test Loss: 0.0332, Test Acc: 0.5205
Epoch [33/250], Train Loss: 0.0259, Train Acc: 0.6138
Epoch [33/250], Test Loss: 0.0333, Test Acc: 0.5342
Epoch [34/250], Train Loss: 0.0260, Train Acc: 0.6189
Epoch [34/250], Test Loss: 0.0328, Test Acc: 0.5479
Epoch [35/250], Train Loss: 0.0256, Train Acc: 0.6419
Epoch [35/250], Test Loss: 0.0325, Test Acc: 0.5342
Epoch [36/250], Train Loss: 0.0248, Train Acc: 0.6496
Epoch [36/250], Test Loss: 0.0338, Test Acc: 0.5068
Epoch [37/250], Train Loss: 0.0242, Train Acc: 0.6419
Epoch [37/250], Test Loss: 0.0317, Test Acc: 0.5479
Epoch [38/250], Train Loss: 0.0259, Train Acc: 0.6394
Epoch [38/250], Test Loss: 0.0327, Test Acc: 0.5342
Epoch [39/250], Train Loss: 0.0233, Train Acc: 0.6547
Epoch [39/250], Test Loss: 0.0322, Test Acc: 0.5479
Epoch [40/250], Train Loss: 0.0235, Train Acc: 0.6777
Epoch [40/250], Test Loss: 0.0317, Test Acc: 0.5342
Epoch [41/250], Train Loss: 0.0228, Train Acc: 0.6675
Epoch [41/250], Test Loss: 0.0313, Test Acc: 0.5616
Epoch [42/250], Train Loss: 0.0232, Train Acc: 0.6650
Epoch [42/250], Test Loss: 0.0317, Test Acc: 0.5753
Epoch [43/250], Train Loss: 0.0227, Train Acc: 0.6752
Epoch [43/250], Test Loss: 0.0303, Test Acc: 0.5616
Epoch [44/250], Train Loss: 0.0226, Train Acc: 0.6726
Epoch [44/250], Test Loss: 0.0307, Test Acc: 0.5342
Epoch [45/250], Train Loss: 0.0221, Train Acc: 0.6854
Epoch [45/250], Test Loss: 0.0306, Test Acc: 0.5753
Epoch [46/250], Train Loss: 0.0212, Train Acc: 0.6880
Epoch [46/250], Test Loss: 0.0305, Test Acc: 0.5342
Epoch [47/250], Train Loss: 0.0227, Train Acc: 0.6905
Epoch [47/250], Test Loss: 0.0303, Test Acc: 0.5616
Epoch [48/250], Train Loss: 0.0251, Train Acc: 0.6471
Epoch [48/250], Test Loss: 0.0306, Test Acc: 0.5753
Epoch [49/250], Train Loss: 0.0220, Train Acc: 0.6880
Epoch [49/250], Test Loss: 0.0309, Test Acc: 0.5616
Epoch [50/250], Train Loss: 0.0224, Train Acc: 0.6803
Epoch [50/250], Test Loss: 0.0299, Test Acc: 0.5753
```

```
Epoch [51/250], Train Loss: 0.0207, Train Acc: 0.6905
Epoch [51/250], Test Loss: 0.0297, Test Acc: 0.5890
Epoch [52/250], Train Loss: 0.0225, Train Acc: 0.6547
Epoch [52/250], Test Loss: 0.0289, Test Acc: 0.5890
Epoch [53/250], Train Loss: 0.0218, Train Acc: 0.6726
Epoch [53/250], Test Loss: 0.0302, Test Acc: 0.5479
Epoch [54/250], Train Loss: 0.0197, Train Acc: 0.7136
Epoch [54/250], Test Loss: 0.0297, Test Acc: 0.5616
Epoch [55/250], Train Loss: 0.0207, Train Acc: 0.6931
Epoch [55/250], Test Loss: 0.0296, Test Acc: 0.5616
Epoch [56/250], Train Loss: 0.0214, Train Acc: 0.6829
Epoch [56/250], Test Loss: 0.0308, Test Acc: 0.5616
Epoch [57/250], Train Loss: 0.0207, Train Acc: 0.6931
Epoch [57/250], Test Loss: 0.0303, Test Acc: 0.5616
Epoch [58/250], Train Loss: 0.0208, Train Acc: 0.6982
Epoch [58/250], Test Loss: 0.0306, Test Acc: 0.5479
Epoch [59/250], Train Loss: 0.0210, Train Acc: 0.7059
Epoch [59/250], Test Loss: 0.0308, Test Acc: 0.5616
Epoch [60/250], Train Loss: 0.0198, Train Acc: 0.7212
Epoch [60/250], Test Loss: 0.0303, Test Acc: 0.5890
Epoch [61/250], Train Loss: 0.0213, Train Acc: 0.6726
Epoch [61/250], Test Loss: 0.0300, Test Acc: 0.5753
Epoch [62/250], Train Loss: 0.0205, Train Acc: 0.7008
Epoch [62/250], Test Loss: 0.0304, Test Acc: 0.5616
Epoch [63/250], Train Loss: 0.0196, Train Acc: 0.7340
Epoch [63/250], Test Loss: 0.0296, Test Acc: 0.5479
Epoch [64/250], Train Loss: 0.0207, Train Acc: 0.6931
Epoch [64/250], Test Loss: 0.0294, Test Acc: 0.5479
Epoch [65/250], Train Loss: 0.0205, Train Acc: 0.6931
Epoch [65/250], Test Loss: 0.0287, Test Acc: 0.5890
Epoch [66/250], Train Loss: 0.0186, Train Acc: 0.7315
Epoch [66/250], Test Loss: 0.0304, Test Acc: 0.5753
Epoch [67/250], Train Loss: 0.0197, Train Acc: 0.7161
Epoch [67/250], Test Loss: 0.0301, Test Acc: 0.5753
Epoch [68/250], Train Loss: 0.0203, Train Acc: 0.7161
Epoch [68/250], Test Loss: 0.0301, Test Acc: 0.5616
Epoch [69/250], Train Loss: 0.0213, Train Acc: 0.7084
Epoch [69/250], Test Loss: 0.0290, Test Acc: 0.5342
Epoch [70/250], Train Loss: 0.0192, Train Acc: 0.7212
Epoch [70/250], Test Loss: 0.0301, Test Acc: 0.5616
Epoch [71/250], Train Loss: 0.0192, Train Acc: 0.7059
Epoch [71/250], Test Loss: 0.0292, Test Acc: 0.5753
Epoch [72/250], Train Loss: 0.0198, Train Acc: 0.7263
Epoch [72/250], Test Loss: 0.0288, Test Acc: 0.5890
Epoch [73/250], Train Loss: 0.0201, Train Acc: 0.7008
Epoch [73/250], Test Loss: 0.0294, Test Acc: 0.5890
Epoch [74/250], Train Loss: 0.0176, Train Acc: 0.7315
Epoch [74/250], Test Loss: 0.0290, Test Acc: 0.5890
Epoch [75/250], Train Loss: 0.0190, Train Acc: 0.7110
Epoch [75/250], Test Loss: 0.0281, Test Acc: 0.5890
```

```
Epoch [76/250], Train Loss: 0.0185, Train Acc: 0.7238
Epoch [76/250], Test Loss: 0.0292, Test Acc: 0.5890
Epoch [77/250], Train Loss: 0.0194, Train Acc: 0.7366
Epoch [77/250], Test Loss: 0.0299, Test Acc: 0.5890
Epoch [78/250], Train Loss: 0.0193, Train Acc: 0.7187
Epoch [78/250], Test Loss: 0.0303, Test Acc: 0.6027
Epoch [79/250], Train Loss: 0.0192, Train Acc: 0.7084
Epoch [79/250], Test Loss: 0.0285, Test Acc: 0.6027
Epoch [80/250], Train Loss: 0.0183, Train Acc: 0.7417
Epoch [80/250], Test Loss: 0.0297, Test Acc: 0.5890
Epoch [81/250], Train Loss: 0.0178, Train Acc: 0.7494
Epoch [81/250], Test Loss: 0.0297, Test Acc: 0.6027
Epoch [82/250], Train Loss: 0.0193, Train Acc: 0.7110
Epoch [82/250], Test Loss: 0.0292, Test Acc: 0.6164
Epoch [83/250], Train Loss: 0.0180, Train Acc: 0.7315
Epoch [83/250], Test Loss: 0.0275, Test Acc: 0.5890
Epoch [84/250], Train Loss: 0.0165, Train Acc: 0.7570
Epoch [84/250], Test Loss: 0.0274, Test Acc: 0.5890
Epoch [85/250], Train Loss: 0.0189, Train Acc: 0.7136
Epoch [85/250], Test Loss: 0.0279, Test Acc: 0.5890
Epoch [86/250], Train Loss: 0.0206, Train Acc: 0.7110
Epoch [86/250], Test Loss: 0.0286, Test Acc: 0.6164
Epoch [87/250], Train Loss: 0.0186, Train Acc: 0.7289
Epoch [87/250], Test Loss: 0.0266, Test Acc: 0.6164
Epoch [88/250], Train Loss: 0.0199, Train Acc: 0.6880
Epoch [88/250], Test Loss: 0.0270, Test Acc: 0.6027
Epoch [89/250], Train Loss: 0.0198, Train Acc: 0.7084
Epoch [89/250], Test Loss: 0.0276, Test Acc: 0.5890
Epoch [90/250], Train Loss: 0.0180, Train Acc: 0.7161
Epoch [90/250], Test Loss: 0.0283, Test Acc: 0.5890
Epoch [91/250], Train Loss: 0.0167, Train Acc: 0.7570
Epoch [91/250], Test Loss: 0.0283, Test Acc: 0.5890
Epoch [92/250], Train Loss: 0.0170, Train Acc: 0.7468
Epoch [92/250], Test Loss: 0.0288, Test Acc: 0.6027
Epoch [93/250], Train Loss: 0.0180, Train Acc: 0.7340
Epoch [93/250], Test Loss: 0.0279, Test Acc: 0.5890
Epoch [94/250], Train Loss: 0.0162, Train Acc: 0.7570
Epoch [94/250], Test Loss: 0.0275, Test Acc: 0.5890
Epoch [95/250], Train Loss: 0.0174, Train Acc: 0.7289
Epoch [95/250], Test Loss: 0.0287, Test Acc: 0.5890
Epoch [96/250], Train Loss: 0.0166, Train Acc: 0.7545
Epoch [96/250], Test Loss: 0.0274, Test Acc: 0.5753
Epoch [97/250], Train Loss: 0.0157, Train Acc: 0.7621
Epoch [97/250], Test Loss: 0.0292, Test Acc: 0.6027
Epoch [98/250], Train Loss: 0.0163, Train Acc: 0.7570
Epoch [98/250], Test Loss: 0.0279, Test Acc: 0.6164
Epoch [99/250], Train Loss: 0.0169, Train Acc: 0.7545
Epoch [99/250], Test Loss: 0.0278, Test Acc: 0.6027
Epoch [100/250], Train Loss: 0.0162, Train Acc: 0.7570
Epoch [100/250], Test Loss: 0.0277, Test Acc: 0.5890
```

```
Epoch [101/250], Train Loss: 0.0160, Train Acc: 0.7621
Epoch [101/250], Test Loss: 0.0295, Test Acc: 0.5890
Epoch [102/250], Train Loss: 0.0172, Train Acc: 0.7059
Epoch [102/250], Test Loss: 0.0283, Test Acc: 0.5890
Epoch [103/250], Train Loss: 0.0165, Train Acc: 0.7519
Epoch [103/250], Test Loss: 0.0284, Test Acc: 0.5890
Epoch [104/250], Train Loss: 0.0174, Train Acc: 0.7315
Epoch [104/250], Test Loss: 0.0283, Test Acc: 0.5753
Epoch [105/250], Train Loss: 0.0161, Train Acc: 0.7775
Epoch [105/250], Test Loss: 0.0280, Test Acc: 0.6027
Epoch [106/250], Train Loss: 0.0165, Train Acc: 0.7519
Epoch [106/250], Test Loss: 0.0273, Test Acc: 0.6027
Epoch [107/250], Train Loss: 0.0165, Train Acc: 0.7417
Epoch [107/250], Test Loss: 0.0277, Test Acc: 0.6164
Epoch [108/250], Train Loss: 0.0160, Train Acc: 0.7621
Epoch [108/250], Test Loss: 0.0308, Test Acc: 0.6027
Epoch [109/250], Train Loss: 0.0150, Train Acc: 0.7826
Epoch [109/250], Test Loss: 0.0287, Test Acc: 0.6027
Epoch [110/250], Train Loss: 0.0177, Train Acc: 0.7519
Epoch [110/250], Test Loss: 0.0292, Test Acc: 0.5890
Epoch [111/250], Train Loss: 0.0164, Train Acc: 0.7596
Epoch [111/250], Test Loss: 0.0303, Test Acc: 0.6027
Epoch [112/250], Train Loss: 0.0162, Train Acc: 0.7545
Epoch [112/250], Test Loss: 0.0299, Test Acc: 0.6027
Epoch [113/250], Train Loss: 0.0170, Train Acc: 0.7289
Epoch [113/250], Test Loss: 0.0311, Test Acc: 0.6164
Epoch [114/250], Train Loss: 0.0158, Train Acc: 0.7570
Epoch [114/250], Test Loss: 0.0297, Test Acc: 0.6164
Epoch [115/250], Train Loss: 0.0155, Train Acc: 0.7596
Epoch [115/250], Test Loss: 0.0299, Test Acc: 0.6164
Epoch [116/250], Train Loss: 0.0157, Train Acc: 0.7647
Epoch [116/250], Test Loss: 0.0312, Test Acc: 0.6164
Epoch [117/250], Train Loss: 0.0162, Train Acc: 0.7621
Epoch [117/250], Test Loss: 0.0293, Test Acc: 0.6438
Epoch [118/250], Train Loss: 0.0154, Train Acc: 0.7673
Epoch [118/250], Test Loss: 0.0293, Test Acc: 0.6301
Epoch [119/250], Train Loss: 0.0151, Train Acc: 0.7775
Epoch [119/250], Test Loss: 0.0289, Test Acc: 0.6301
Epoch [120/250], Train Loss: 0.0153, Train Acc: 0.7724
Epoch [120/250], Test Loss: 0.0309, Test Acc: 0.6027
Epoch [121/250], Train Loss: 0.0154, Train Acc: 0.7698
Epoch [121/250], Test Loss: 0.0299, Test Acc: 0.6027
Epoch [122/250], Train Loss: 0.0148, Train Acc: 0.7852
Epoch [122/250], Test Loss: 0.0296, Test Acc: 0.6164
Epoch [123/250], Train Loss: 0.0155, Train Acc: 0.7621
Epoch [123/250], Test Loss: 0.0287, Test Acc: 0.6027
Epoch [124/250], Train Loss: 0.0150, Train Acc: 0.7596
Epoch [124/250], Test Loss: 0.0297, Test Acc: 0.6164
Epoch [125/250], Train Loss: 0.0155, Train Acc: 0.7468
Epoch [125/250], Test Loss: 0.0309, Test Acc: 0.5890
```

```
Epoch [126/250], Train Loss: 0.0153, Train Acc: 0.7775
Epoch [126/250], Test Loss: 0.0309, Test Acc: 0.5890
Epoch [127/250], Train Loss: 0.0165, Train Acc: 0.7545
Epoch [127/250], Test Loss: 0.0302, Test Acc: 0.5753
Epoch [128/250], Train Loss: 0.0148, Train Acc: 0.7749
Epoch [128/250], Test Loss: 0.0284, Test Acc: 0.6301
Epoch [129/250], Train Loss: 0.0142, Train Acc: 0.7852
Epoch [129/250], Test Loss: 0.0285, Test Acc: 0.6301
Epoch [130/250], Train Loss: 0.0150, Train Acc: 0.7647
Epoch [130/250], Test Loss: 0.0282, Test Acc: 0.6164
Epoch [131/250], Train Loss: 0.0168, Train Acc: 0.7749
Epoch [131/250], Test Loss: 0.0292, Test Acc: 0.6027
Epoch [132/250], Train Loss: 0.0154, Train Acc: 0.7673
Epoch [132/250], Test Loss: 0.0272, Test Acc: 0.6027
Epoch [133/250], Train Loss: 0.0143, Train Acc: 0.7698
Epoch [133/250], Test Loss: 0.0271, Test Acc: 0.6164
Epoch [134/250], Train Loss: 0.0145, Train Acc: 0.7673
Epoch [134/250], Test Loss: 0.0260, Test Acc: 0.6301
Epoch [135/250], Train Loss: 0.0146, Train Acc: 0.7852
Epoch [135/250], Test Loss: 0.0296, Test Acc: 0.6164
Epoch [136/250], Train Loss: 0.0148, Train Acc: 0.7877
Epoch [136/250], Test Loss: 0.0281, Test Acc: 0.6301
Epoch [137/250], Train Loss: 0.0154, Train Acc: 0.7749
Epoch [137/250], Test Loss: 0.0273, Test Acc: 0.6301
Epoch [138/250], Train Loss: 0.0169, Train Acc: 0.7673
Epoch [138/250], Test Loss: 0.0276, Test Acc: 0.6301
Epoch [139/250], Train Loss: 0.0151, Train Acc: 0.7826
Epoch [139/250], Test Loss: 0.0265, Test Acc: 0.6712
Epoch [140/250], Train Loss: 0.0146, Train Acc: 0.7903
Epoch [140/250], Test Loss: 0.0290, Test Acc: 0.6438
Epoch [141/250], Train Loss: 0.0144, Train Acc: 0.7903
Epoch [141/250], Test Loss: 0.0273, Test Acc: 0.6301
Epoch [142/250], Train Loss: 0.0144, Train Acc: 0.7698
Epoch [142/250], Test Loss: 0.0295, Test Acc: 0.6438
Epoch [143/250], Train Loss: 0.0172, Train Acc: 0.7519
Epoch [143/250], Test Loss: 0.0291, Test Acc: 0.6164
Epoch [144/250], Train Loss: 0.0160, Train Acc: 0.7442
Epoch [144/250], Test Loss: 0.0315, Test Acc: 0.6301
Epoch [145/250], Train Loss: 0.0157, Train Acc: 0.7647
Epoch [145/250], Test Loss: 0.0282, Test Acc: 0.6301
Epoch [146/250], Train Loss: 0.0152, Train Acc: 0.7801
Epoch [146/250], Test Loss: 0.0278, Test Acc: 0.6301
Epoch [147/250], Train Loss: 0.0159, Train Acc: 0.7724
Epoch [147/250], Test Loss: 0.0267, Test Acc: 0.6438
Epoch [148/250], Train Loss: 0.0143, Train Acc: 0.7980
Epoch [148/250], Test Loss: 0.0287, Test Acc: 0.6301
Epoch [149/250], Train Loss: 0.0139, Train Acc: 0.7826
Epoch [149/250], Test Loss: 0.0269, Test Acc: 0.6575
Epoch [150/250], Train Loss: 0.0125, Train Acc: 0.8082
Epoch [150/250], Test Loss: 0.0275, Test Acc: 0.6301
```

```
Epoch [151/250], Train Loss: 0.0124, Train Acc: 0.8286
Epoch [151/250], Test Loss: 0.0319, Test Acc: 0.6438
Epoch [152/250], Train Loss: 0.0141, Train Acc: 0.7903
Epoch [152/250], Test Loss: 0.0290, Test Acc: 0.6301
Epoch [153/250], Train Loss: 0.0153, Train Acc: 0.7673
Epoch [153/250], Test Loss: 0.0274, Test Acc: 0.6575
Epoch [154/250], Train Loss: 0.0149, Train Acc: 0.7877
Epoch [154/250], Test Loss: 0.0289, Test Acc: 0.6575
Epoch [155/250], Train Loss: 0.0136, Train Acc: 0.7903
Epoch [155/250], Test Loss: 0.0291, Test Acc: 0.6438
Epoch [156/250], Train Loss: 0.0139, Train Acc: 0.7749
Epoch [156/250], Test Loss: 0.0293, Test Acc: 0.6438
Epoch [157/250], Train Loss: 0.0140, Train Acc: 0.7903
Epoch [157/250], Test Loss: 0.0325, Test Acc: 0.6164
Epoch [158/250], Train Loss: 0.0145, Train Acc: 0.7826
Epoch [158/250], Test Loss: 0.0304, Test Acc: 0.6301
Epoch [159/250], Train Loss: 0.0132, Train Acc: 0.8005
Epoch [159/250], Test Loss: 0.0311, Test Acc: 0.6301
Epoch [160/250], Train Loss: 0.0140, Train Acc: 0.7775
Epoch [160/250], Test Loss: 0.0302, Test Acc: 0.6438
Epoch [161/250], Train Loss: 0.0125, Train Acc: 0.8107
Epoch [161/250], Test Loss: 0.0317, Test Acc: 0.6438
Epoch [162/250], Train Loss: 0.0133, Train Acc: 0.8005
Epoch [162/250], Test Loss: 0.0302, Test Acc: 0.6712
Epoch [163/250], Train Loss: 0.0128, Train Acc: 0.7954
Epoch [163/250], Test Loss: 0.0284, Test Acc: 0.6575
Epoch [164/250], Train Loss: 0.0156, Train Acc: 0.7749
Epoch [164/250], Test Loss: 0.0280, Test Acc: 0.6575
Epoch [165/250], Train Loss: 0.0128, Train Acc: 0.8056
Epoch [165/250], Test Loss: 0.0259, Test Acc: 0.6575
Epoch [166/250], Train Loss: 0.0136, Train Acc: 0.8031
Epoch [166/250], Test Loss: 0.0266, Test Acc: 0.6438
Epoch [167/250], Train Loss: 0.0129, Train Acc: 0.7954
Epoch [167/250], Test Loss: 0.0250, Test Acc: 0.6575
Epoch [168/250], Train Loss: 0.0144, Train Acc: 0.7801
Epoch [168/250], Test Loss: 0.0256, Test Acc: 0.6575
Epoch [169/250], Train Loss: 0.0128, Train Acc: 0.8235
Epoch [169/250], Test Loss: 0.0261, Test Acc: 0.6438
Epoch [170/250], Train Loss: 0.0137, Train Acc: 0.7801
Epoch [170/250], Test Loss: 0.0254, Test Acc: 0.6575
Epoch [171/250], Train Loss: 0.0138, Train Acc: 0.7903
Epoch [171/250], Test Loss: 0.0251, Test Acc: 0.6575
Epoch [172/250], Train Loss: 0.0132, Train Acc: 0.7826
Epoch [172/250], Test Loss: 0.0252, Test Acc: 0.6849
Epoch [173/250], Train Loss: 0.0143, Train Acc: 0.7724
Epoch [173/250], Test Loss: 0.0253, Test Acc: 0.6301
Epoch [174/250], Train Loss: 0.0131, Train Acc: 0.7775
Epoch [174/250], Test Loss: 0.0247, Test Acc: 0.6712
Epoch [175/250], Train Loss: 0.0134, Train Acc: 0.8159
Epoch [175/250], Test Loss: 0.0245, Test Acc: 0.6575
```

```
Epoch [176/250], Train Loss: 0.0135, Train Acc: 0.7980
Epoch [176/250], Test Loss: 0.0248, Test Acc: 0.6712
Epoch [177/250], Train Loss: 0.0143, Train Acc: 0.7826
Epoch [177/250], Test Loss: 0.0251, Test Acc: 0.6849
Epoch [178/250], Train Loss: 0.0133, Train Acc: 0.7928
Epoch [178/250], Test Loss: 0.0254, Test Acc: 0.6849
Epoch [179/250], Train Loss: 0.0127, Train Acc: 0.8005
Epoch [179/250], Test Loss: 0.0249, Test Acc: 0.6438
Epoch [180/250], Train Loss: 0.0148, Train Acc: 0.7801
Epoch [180/250], Test Loss: 0.0258, Test Acc: 0.6438
Epoch [181/250], Train Loss: 0.0134, Train Acc: 0.7903
Epoch [181/250], Test Loss: 0.0252, Test Acc: 0.6849
Epoch [182/250], Train Loss: 0.0128, Train Acc: 0.8107
Epoch [182/250], Test Loss: 0.0238, Test Acc: 0.6986
Epoch [183/250], Train Loss: 0.0126, Train Acc: 0.8082
Epoch [183/250], Test Loss: 0.0250, Test Acc: 0.6712
Epoch [184/250], Train Loss: 0.0137, Train Acc: 0.8005
Epoch [184/250], Test Loss: 0.0237, Test Acc: 0.6575
Epoch [185/250], Train Loss: 0.0137, Train Acc: 0.7954
Epoch [185/250], Test Loss: 0.0238, Test Acc: 0.7260
Epoch [186/250], Train Loss: 0.0144, Train Acc: 0.7928
Epoch [186/250], Test Loss: 0.0235, Test Acc: 0.6438
Epoch [187/250], Train Loss: 0.0130, Train Acc: 0.7954
Epoch [187/250], Test Loss: 0.0239, Test Acc: 0.6849
Epoch [188/250], Train Loss: 0.0143, Train Acc: 0.7724
Epoch [188/250], Test Loss: 0.0247, Test Acc: 0.6712
Epoch [189/250], Train Loss: 0.0152, Train Acc: 0.7570
Epoch [189/250], Test Loss: 0.0237, Test Acc: 0.6986
Epoch [190/250], Train Loss: 0.0122, Train Acc: 0.8133
Epoch [190/250], Test Loss: 0.0240, Test Acc: 0.6575
Epoch [191/250], Train Loss: 0.0137, Train Acc: 0.7801
Epoch [191/250], Test Loss: 0.0233, Test Acc: 0.6712
Epoch [192/250], Train Loss: 0.0137, Train Acc: 0.7877
Epoch [192/250], Test Loss: 0.0238, Test Acc: 0.6712
Epoch [193/250], Train Loss: 0.0145, Train Acc: 0.8005
Epoch [193/250], Test Loss: 0.0237, Test Acc: 0.6712
Epoch [194/250], Train Loss: 0.0131, Train Acc: 0.8133
Epoch [194/250], Test Loss: 0.0243, Test Acc: 0.6849
Epoch [195/250], Train Loss: 0.0127, Train Acc: 0.8159
Epoch [195/250], Test Loss: 0.0231, Test Acc: 0.6849
Epoch [196/250], Train Loss: 0.0119, Train Acc: 0.8082
Epoch [196/250], Test Loss: 0.0239, Test Acc: 0.6712
Epoch [197/250], Train Loss: 0.0130, Train Acc: 0.8056
Epoch [197/250], Test Loss: 0.0248, Test Acc: 0.6575
Epoch [198/250], Train Loss: 0.0140, Train Acc: 0.7852
Epoch [198/250], Test Loss: 0.0251, Test Acc: 0.6575
Epoch [199/250], Train Loss: 0.0139, Train Acc: 0.7775
Epoch [199/250], Test Loss: 0.0239, Test Acc: 0.6575
Epoch [200/250], Train Loss: 0.0136, Train Acc: 0.7775
Epoch [200/250], Test Loss: 0.0258, Test Acc: 0.6438
```

```
Epoch [201/250], Train Loss: 0.0128, Train Acc: 0.7980
Epoch [201/250], Test Loss: 0.0241, Test Acc: 0.6301
Epoch [202/250], Train Loss: 0.0132, Train Acc: 0.8184
Epoch [202/250], Test Loss: 0.0251, Test Acc: 0.6438
Epoch [203/250], Train Loss: 0.0132, Train Acc: 0.8005
Epoch [203/250], Test Loss: 0.0261, Test Acc: 0.6438
Epoch [204/250], Train Loss: 0.0126, Train Acc: 0.8159
Epoch [204/250], Test Loss: 0.0265, Test Acc: 0.6438
Epoch [205/250], Train Loss: 0.0120, Train Acc: 0.8261
Epoch [205/250], Test Loss: 0.0268, Test Acc: 0.6438
Epoch [206/250], Train Loss: 0.0121, Train Acc: 0.8005
Epoch [206/250], Test Loss: 0.0259, Test Acc: 0.6438
Epoch [207/250], Train Loss: 0.0122, Train Acc: 0.8133
Epoch [207/250], Test Loss: 0.0261, Test Acc: 0.6438
Epoch [208/250], Train Loss: 0.0132, Train Acc: 0.7903
Epoch [208/250], Test Loss: 0.0262, Test Acc: 0.6712
Epoch [209/250], Train Loss: 0.0125, Train Acc: 0.8056
Epoch [209/250], Test Loss: 0.0253, Test Acc: 0.6438
Epoch [210/250], Train Loss: 0.0131, Train Acc: 0.8133
Epoch [210/250], Test Loss: 0.0265, Test Acc: 0.6301
Epoch [211/250], Train Loss: 0.0121, Train Acc: 0.8363
Epoch [211/250], Test Loss: 0.0252, Test Acc: 0.6575
Epoch [212/250], Train Loss: 0.0130, Train Acc: 0.8159
Epoch [212/250], Test Loss: 0.0252, Test Acc: 0.6301
Epoch [213/250], Train Loss: 0.0136, Train Acc: 0.8031
Epoch [213/250], Test Loss: 0.0267, Test Acc: 0.6438
Epoch [214/250], Train Loss: 0.0115, Train Acc: 0.8184
Epoch [214/250], Test Loss: 0.0272, Test Acc: 0.6301
Epoch [215/250], Train Loss: 0.0126, Train Acc: 0.8082
Epoch [215/250], Test Loss: 0.0269, Test Acc: 0.6438
Epoch [216/250], Train Loss: 0.0115, Train Acc: 0.8389
Epoch [216/250], Test Loss: 0.0252, Test Acc: 0.6575
Epoch [217/250], Train Loss: 0.0126, Train Acc: 0.8107
Epoch [217/250], Test Loss: 0.0259, Test Acc: 0.6575
Epoch [218/250], Train Loss: 0.0131, Train Acc: 0.8184
Epoch [218/250], Test Loss: 0.0259, Test Acc: 0.6301
Epoch [219/250], Train Loss: 0.0134, Train Acc: 0.8133
Epoch [219/250], Test Loss: 0.0280, Test Acc: 0.6438
Epoch [220/250], Train Loss: 0.0123, Train Acc: 0.8133
Epoch [220/250], Test Loss: 0.0250, Test Acc: 0.6301
Epoch [221/250], Train Loss: 0.0114, Train Acc: 0.8491
Epoch [221/250], Test Loss: 0.0255, Test Acc: 0.6438
Epoch [222/250], Train Loss: 0.0126, Train Acc: 0.8184
Epoch [222/250], Test Loss: 0.0272, Test Acc: 0.6438
Epoch [223/250], Train Loss: 0.0125, Train Acc: 0.8056
Epoch [223/250], Test Loss: 0.0271, Test Acc: 0.6575
Epoch [224/250], Train Loss: 0.0155, Train Acc: 0.8005
Epoch [224/250], Test Loss: 0.0266, Test Acc: 0.6164
Epoch [225/250], Train Loss: 0.0124, Train Acc: 0.8107
Epoch [225/250], Test Loss: 0.0265, Test Acc: 0.6575
```

```
Epoch [226/250], Train Loss: 0.0127, Train Acc: 0.7954
Epoch [226/250], Test Loss: 0.0286, Test Acc: 0.6301
Epoch [227/250], Train Loss: 0.0125, Train Acc: 0.8031
Epoch [227/250], Test Loss: 0.0289, Test Acc: 0.6301
Epoch [228/250], Train Loss: 0.0129, Train Acc: 0.8031
Epoch [228/250], Test Loss: 0.0258, Test Acc: 0.6575
Epoch [229/250], Train Loss: 0.0120, Train Acc: 0.8235
Epoch [229/250], Test Loss: 0.0261, Test Acc: 0.6301
Epoch [230/250], Train Loss: 0.0115, Train Acc: 0.8363
Epoch [230/250], Test Loss: 0.0265, Test Acc: 0.6438
Epoch [231/250], Train Loss: 0.0121, Train Acc: 0.8363
Epoch [231/250], Test Loss: 0.0271, Test Acc: 0.6301
Epoch [232/250], Train Loss: 0.0114, Train Acc: 0.8338
Epoch [232/250], Test Loss: 0.0272, Test Acc: 0.6849
Epoch [233/250], Train Loss: 0.0145, Train Acc: 0.7775
Epoch [233/250], Test Loss: 0.0273, Test Acc: 0.6712
Epoch [234/250], Train Loss: 0.0122, Train Acc: 0.8210
Epoch [234/250], Test Loss: 0.0274, Test Acc: 0.6986
Epoch [235/250], Train Loss: 0.0115, Train Acc: 0.8210
Epoch [235/250], Test Loss: 0.0282, Test Acc: 0.6849
Epoch [236/250], Train Loss: 0.0128, Train Acc: 0.8159
Epoch [236/250], Test Loss: 0.0262, Test Acc: 0.6986
Epoch [237/250], Train Loss: 0.0117, Train Acc: 0.8056
Epoch [237/250], Test Loss: 0.0267, Test Acc: 0.6986
Epoch [238/250], Train Loss: 0.0127, Train Acc: 0.8056
Epoch [238/250], Test Loss: 0.0262, Test Acc: 0.6849
Epoch [239/250], Train Loss: 0.0109, Train Acc: 0.8593
Epoch [239/250], Test Loss: 0.0270, Test Acc: 0.6849
Epoch [240/250], Train Loss: 0.0121, Train Acc: 0.8184
Epoch [240/250], Test Loss: 0.0295, Test Acc: 0.6712
Epoch [241/250], Train Loss: 0.0140, Train Acc: 0.7980
Epoch [241/250], Test Loss: 0.0287, Test Acc: 0.6301
Epoch [242/250], Train Loss: 0.0122, Train Acc: 0.8184
Epoch [242/250], Test Loss: 0.0267, Test Acc: 0.6712
Epoch [243/250], Train Loss: 0.0133, Train Acc: 0.8082
Epoch [243/250], Test Loss: 0.0271, Test Acc: 0.6301
Epoch [244/250], Train Loss: 0.0124, Train Acc: 0.8235
Epoch [244/250], Test Loss: 0.0253, Test Acc: 0.6438
Epoch [245/250], Train Loss: 0.0110, Train Acc: 0.8184
Epoch [245/250], Test Loss: 0.0246, Test Acc: 0.6849
Epoch [246/250], Train Loss: 0.0114, Train Acc: 0.8184
Epoch [246/250], Test Loss: 0.0232, Test Acc: 0.6849
Epoch [247/250], Train Loss: 0.0134, Train Acc: 0.7903
Epoch [247/250], Test Loss: 0.0238, Test Acc: 0.6986
Epoch [248/250], Train Loss: 0.0135, Train Acc: 0.8031
Epoch [248/250], Test Loss: 0.0241, Test Acc: 0.6986
Epoch [249/250], Train Loss: 0.0110, Train Acc: 0.8414
Epoch [249/250], Test Loss: 0.0254, Test Acc: 0.7123
Epoch [250/250], Train Loss: 0.0120, Train Acc: 0.8184
```

```
Epoch [250/250], Test Loss: 0.0238, Test Acc: 0.6986
Best Test Accuracy: 0.7260
```

```python
def load_trained_model(model_path, input_size, hidden_size1,
hidden_size2, output_size):
    model = ComplexNeuralNetwork(input_size, hidden_size1,
hidden_size2, output_size)
    model.load_state_dict(torch.load(model_path))
    model.eval()
    return model

# Function to preprocess the input sentence
def preprocess_sentence(sentence, words):
    sentence_words = sentence.lower().split()
    sentence_words = [word for word in sentence_words if word in
words]
    return sentence_words

# Function to convert the preprocessed sentence into a feature vector
def sentence_to_features(sentence_words, words):
    features = [1 if word in sentence_words else 0 for word in words]
    return torch.tensor(features).float().unsqueeze(0)

# Function to generate a response using the trained model
def generate_response(sentence, model, words, classes, intents):
    sentence_words = preprocess_sentence(sentence, words)
    if len(sentence_words) == 0:
        return "I'm sorry, but I don't understand. Can you please
rephrase or provide more information?"

    features = sentence_to_features(sentence_words, words)
    with torch.no_grad():
        outputs = model(features)

    probabilities, predicted_class = torch.max(outputs, dim=1)
    confidence = probabilities.item()
    predicted_tag = classes[predicted_class.item()]

    if confidence > 0.3:
        for intent in intents['intents']:
            if intent['tag'] == predicted_tag:
                return random.choice(intent['responses'])

    return "I'm sorry, but I'm not sure how to respond to that."

# Load the trained model
model_path = 'best_model.pth'
input_size = len(word_list)
hidden_size1 = 50
hidden_size2 = 30
```

```
output_size = len(class_list)
model = load_trained_model(model_path, input_size, hidden_size1,
hidden_size2, output_size)

import os

# Load the trained model
model_path = 'best_model.pth'  # Update the model path if needed
input_size = len(word_list)
hidden_size1 = 50
hidden_size2 = 30
output_size = len(class_list)
model = load_trained_model(model_path, input_size, hidden_size1,
hidden_size2, output_size)

class ChatBot:
  def __init__(self):
    self.user_models = {}
    self.load_user_models()

  def load_user_models(self):
    if os.path.exists("user_models.json"):
      with open("user_models.json", "r") as file:
        self.user_models = json.load(file)

  def save_user_models(self):
    with open("user_models.json", "w") as file:
      json.dump(self.user_models, file, indent=4)

  def get_user_model(self, user_id):
    if user_id not in self.user_models:
      self.user_models[user_id] = {"name": "", "likes": [],
"dislikes": [], "conversations": []}
    return self.user_models[user_id]

  def update_user_model(self, user_id, data):
    self.user_models[user_id].update(data)

  def chat(self):
    user_id = input('Hello! I am a chatbot. Before we start, tell me
your name. Type "quit" to exit.', )
    user_model = self.get_user_model(user_id)
    if not user_model["name"]:
      user_name = input("User Name not found. Creating a new User. \n
Nice to meet you, {}. What should I call you? ".format(user_id))
      user_model["name"] = user_name
      user_likes = input("Tell me some of the things you like. \n")
      user_model["likes"]. append(user_likes)
      user_dislikes = input("Tell me some of the things you don't
like. \n")
```

```
      user_model["dislikes"]. append(user_dislikes)
      print("Thankyou for sharing that with me, {}! How can I help you
today?".format(user_model["name"]))
    else:
      print("Greetings {}! How can I help you
today?".format(user_model["name"]))

    while True:
        user_input = input('> ')
        if user_input.lower() == 'quit':
          # Save the user model
          self.save_user_models()
          break

        # Generate a response to the user's input
        response = generate_response(user_input, model, word_list,
class_list, intents)
        # Update the response in the user model if there are previous
conversations
        if user_model["conversations"]:
            user_model["conversations"][-1]["response"] = response

        # Append the current conversation to the user model
        user_model["conversations"].append({"input": user_input,
"response": response})


        # Print the response
        print(response)

if __name__ == "__main__":

    chatbot = ChatBot()
    chatbot.chat()
```

```
Hello! I am a chatbot. Before we start, tell me your name. Type "quit"
to exit.Shubhrangshu Bose
User Name not found. Creating a new User.
 Nice to meet you, Shubhrangshu Bose. What should I call you? Bose
Tell me some of the things you like.
Basketball, Poker, Blackjack
Tell me some of the things you don't like.
Swimming
Thankyou for sharing that with me, Bose! How can I help you today?
> Who are you
I'm Pandora. I am a conversational agent designed to mimic a
therapist. So how are you feeling today?
> Tell me a fact about mental health
Depression is the leading cause of disability worldwide.
> oh
```

```
Let's discuss further why you're feeling this way.
> nothing much
Oh I see. Do you want to talk about something?
> no
I'm listening. Please go on.
> good night
Good night. Get some proper sleep
> thanks
My pleasure
> quit
```