ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

ΣΧΟΛΗ ΤΕΧΝΟΛΟΓΙΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΕΠΙΚΟΙΝΩΝΙΩΝ

ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ

ΕΡΓΑΣΙΑ

ΕΠΕΞΕΡΓΑΣΙΑ ΦΥΣΙΚΗΣ ΓΛΩΣΣΑΣ

Ακαδημαϊκό έτος 2021-2022

ΕΠΙΒΛΕΠΩΝ ΚΑΘΗΓΗΤΗΣ: ΘΕΜΗΣ ΠΑΝΑΓΙΩΤΟΠΟΥΛΟΣ

*ΒΙΤΑΚΗΣ ΑΘΑΝΑΣΙΟΣ - **Π19247***

*ΑΥΓΕΡΙΝΟΣ ΧΡΗΣΤΟΣ - **Π19020***

*ΠΑΝΑΓΙΩΤΟΠΟΥΛΟΣ ΔΗΜΗΤΡΙΟΣ - **Π19130***

Πρόλογος

Στην παρούσα πρότυπη εργασία για το μάθημα της **Επεξεργασίας Φυσικής Γλώσσας** 6ου εξαμήνου του τμήματος Πληροφορικής του Πανεπιστημίου Πειραιώς επιλέχθηκε ως θέμα η ανάπτυξη ενός ευφυή συστήματος με σκοπό την αλληλεπίδραση του χρήστη σε φυσική γλώσσα με ένα φανταστικό πρόσωπο από την σειρά βιβλίων και βιντεοπαιχνιδιών ονόματι **The Witcher** ή στην ελληνική μετάφραση **Ο Γητευτής.**

Το παρών θέμα της εργασίας αποτελεί σε κύριο λόγο αντικείμενο τεχνητής νοημοσύνης(**AI-Artificial Intelligence**) και προσομοίωση συζήτησης με έναν χρήστη σε φυσική γλώσσα(**Natural Language Processing**) , θέματα που και τα δύο είναι επίκαιρα με έναν από τους βασικούς στόχους να βελτιώσουν την αλληλεπίδραση χρήστη – υπολογιστή ,να την κάνουν περισσότερη δια δραστική και ενδιαφέρουσα και να την προσαρμόσουν απάνω στις ανάγκες του χρήστη.

Λέξεις κλειδιά

Ευφυές σύστημα, τεχνητή νοημοσύνη, φυσική γλώσσα (Natural Language Processing), αλληλεπίδραση ανθρώπου-υπολογιστή, AI-Artificial Intelligence,Machine Learning(ML),Τεχνητή Μάθηση, Νευρωνικά δίκτυα.

Βιτάκης Αθανάσιος/ Π19247
Αυγερινός Χρήστος/ Π19020
Παναγιωτόπουλος Δημήτριος /Π19130

Πίνακας περιεχομένων

Βιτάκης Αθανάσιος/ Π19247
Αυγερινός Χρήστος/ Π19020
Παναγιωτόπουλος Δημήτριος /Π19130

1.Πληροφορίες σχετικά με την τεχνολογία chatbot

Τι είναι το chatbot

Στο πιο βασικό επίπεδο, ένα chatbot είναι ένα πρόγραμμα υπολογιστή που προσομοιώνει και επεξεργάζεται ανθρώπινη συνομιλία (είτε γραπτή είτε προφορική), επιτρέποντας στους ανθρώπους να αλληλεπιδρούν με ψηφιακές συσκευές σαν να επικοινωνούν με ένα πραγματικό άτομο. Τα chatbots μπορεί να είναι τόσο απλά όσο τα στοιχειώδη προγράμματα που απαντούν σε ένα απλό ερώτημα με μια απάντηση μίας γραμμής ή τόσο εξελιγμένα όσο οι ψηφιακοί βοηθοί που μαθαίνουν και εξελίσσονται για να προσφέρουν αυξανόμενα επίπεδα εξατομίκευσης καθώς συλλέγουν και επεξεργάζονται πληροφορίες.

Πώς λειτουργεί ένα chatbot

- Δηλωτικά(Task -oriented chatbots):

  είναι προγράμματα ενός σκοπού που επικεντρώνονται στην εκτέλεση μιας λειτουργίας. Χρησιμοποιώντας κανόνες, NLP και πολύ λίγο Machine Learning, δημιουργούν αυτοματοποιημένες αλλά συνομιλητικές απαντήσεις σε ερωτήματα χρηστών. Οι αλληλεπιδράσεις με αυτά τα chatbots είναι εξαιρετικά συγκεκριμένες και δομημένες και είναι πιο εφαρμόσιμες σε λειτουργίες υποστήριξης. Τα chatbot προσανατολισμένα σε εργασίες μπορούν να χειριστούν συνήθεις ερωτήσεις, όπως ερωτήματα σχετικά με ώρες εργασίας ή απλές συναλλαγές που δεν περιλαμβάνουν μια ποικιλία μεταβλητών. Αν και χρησιμοποιούν το NLP, ώστε οι τελικοί χρήστες να μπορούν να το βιώσουν με συνομιλητικό τρόπο, οι δυνατότητές τους είναι αρκετά βασικές. Αυτά είναι επί του παρόντος τα πιο συχνά χρησιμοποιούμενα chatbot.

- Ομιλητικά(Data-driven and predictive chatbots)

  Συχνά αναφέρονται ως virtual assistants ή digital assistants ,και είναι πολύ πιο εξελιγμένοι, διαδραστικοί και εξατομικευμένοι από τα δηλωτικά chatbots. Αυτά τα chatbots έχουν επίγνωση των συμφραζομένων και αξιοποιούν την κατανόηση φυσικής γλώσσας (NLU),

NLP και Machine Learning για να μαθαίνουν καθώς προχωρούν. Εφαρμόζουν προγνωστική νοημοσύνη και αναλυτικά στοιχεία για να επιτρέψουν την εξατομίκευση με βάση τα προφίλ χρηστών και τη συμπεριφορά του παρελθόντος χρήστη. Οι ψηφιακοί βοηθοί μπορούν να μάθουν τις προτιμήσεις ενός χρήστη με την πάροδο του χρόνου, να παρέχουν συστάσεις και ακόμη και να προβλέψουν τις ανάγκες. Εκτός από την παρακολούθηση δεδομένων και την πρόθεση, μπορούν να ξεκινήσουν συνομιλίες. Το Siri της Apple και το Alexa της Amazon είναι παραδείγματα προγνωστικών chatbot με γνώμονα τον καταναλωτή, με γνώμονα τα δεδομένα.

Η παρακάτω εργασία ανήκει κυρίως στην κατηγορία Data-driven and predictive chatbots

Γιατί δημιουργήθηκαν τα chatbot

Η ψηφιοποίηση μετατρέπει την κοινωνία σε πληθυσμό «προς τα κινητά». Καθώς οι εφαρμογές ανταλλαγής μηνυμάτων αυξάνονται σε δημοτικότητα, τα chatbots παίζουν όλο και περισσότερο σημαντικό ρόλο σε αυτόν τον μετασχηματισμό που βασίζεται στην κινητικότητα. Τα έξυπνα chatbot συνομιλίας είναι συχνά διεπαφές για εφαρμογές για κινητές συσκευές και αλλάζουν τον τρόπο αλληλεπίδρασης επιχειρήσεων και πελατών.

Τα chatbot επιτρέπουν στις επιχειρήσεις να συνδέονται με τους πελάτες με προσωπικό τρόπο χωρίς το κόστος των ανθρώπινων εκπροσώπων. Για παράδειγμα, πολλές από τις ερωτήσεις ή τα ζητήματα που έχουν οι πελάτες είναι κοινές και απαντώνται εύκολα. Γι' αυτό οι εταιρείες δημιουργούν συχνές ερωτήσεις και οδηγούς αντιμετώπισης προβλημάτων. Τα chatbots παρέχουν μια προσωπική εναλλακτική σε γραπτές Συχνές ερωτήσεις ή οδηγό και μπορούν ακόμη και να προσδιορίσουν ερωτήσεις, συμπεριλαμβανομένης της παράδοσης ενός ζητήματος πελάτη σε ζωντανό άτομο, εάν το ζήτημα γίνει πολύ περίπλοκο για να επιλύσει το chatbot. Τα chatbots έχουν γίνει δημοφιλή ως εξοικονόμηση χρόνου και χρημάτων για τις επιχειρήσεις και ως πρόσθετη ευκολία για τους πελάτες.

Η εξέλιξη των chatbot

Η προέλευση του chatbot βρίσκεται αναμφισβήτητα στο όραμα του Alan Turing της δεκαετίας του 1950 για τις ευφυείς μηχανές. Η τεχνητή νοημοσύνη, το θεμέλιο για τα chatbots, έχει προχωρήσει από τότε και περιλαμβάνει υπερέξυπνους υπερυπολογιστές όπως η IBM Watson.

Το αρχικό chatbot ήταν το τηλεφωνικό δέντρο, το οποίο οδήγησε τους πελάτες που εισέρχονταν στο τηλέφωνο σε μια συχνά δυσκίνητη και απογοητευτική διαδρομή να επιλέγουν τη μία επιλογή μετά την άλλη για να περάσουν από ένα αυτοματοποιημένο μοντέλο εξυπηρέτησης πελατών. Οι βελτιώσεις στην τεχνολογία και η αυξανόμενη πολυπλοκότητα των AI, ML και NLP εξέλιξαν αυτό το μοντέλο σε αναδυόμενες, ζωντανές συνομιλίες στην οθόνη. Και το εξελικτικό ταξίδι συνεχίστηκε.

Με τους σημερινούς ψηφιακούς βοηθούς, οι επιχειρήσεις μπορούν να κλιμακώσουν την τεχνητή νοημοσύνη για να παρέχουν πολύ πιο βολικές και αποτελεσματικές αλληλεπιδράσεις μεταξύ εταιρειών και πελατών – απευθείας από τις ψηφιακές συσκευές των πελατών.

2.Δημιουργία chatbot

Εισαγωγή

Στην παρακάτω εργασία έχει αναπτυχθεί ένα chatbot χρησιμοποιώντας τεχνικές βαθιάς μάθησης **(deep learning)** σε γλώσσα προγραμματισμού Python. Συγκεκριμένα εκπαιδεύουμε το πρόγραμμα με πιθανές ερωτήσεις και απαντήσεις για ένα συγκεκριμένο θέμα και ύστερα το πρόγραμμα αναγνωρίζει κατά πόσο μια ερώτηση από τον χρήστη ταιριάζει με τις ερωτήσεις που του προμηθεύσαμε, τέλος αν το ποσοστό αντιστοιχίας ξεπερνάει το 75% τότε του εμφανίζουμε μια τυχαία απάντηση από τις διαθέσιμες του ερωτήματος που έγινε αντιστοιχία.

Γενικά το πρόγραμμα ακολουθεί τον παρακάτω αλγόριθμο διαδικασιών:

1. Λάβετε κάποια στοιχεία από τον χρήστη
2. Μετατρέψτε το σε ένα σάκο με λέξεις
3. Λάβετε μια πρόβλεψη από το μοντέλο
4. Βρείτε την πιο πιθανή τάξη
5. Επιλέξτε μια απάντηση από αυτήν την τάξη

Βιτάκης Αθανάσιος/ Π19247
Αυγερινός Χρήστος/ Π19020
Παναγιωτόπουλος Δημήτριος /Π19130

Αρχεία

Main.py: Το αρχείο αυτό αποτελεί τον πηγαίο κώδικα του κύριου project.

discordConnect.py: Το αρχείο αυτό αποτελεί το Main.py με διαφοροποιήσεις ώστε να δουλεύει σαν discord bot.

Object.json: Το αρχείο αυτό περιέχει τα δεδομένα με τα οποία εκπαιδεύουμε το chatbot.

Ambience.mp3: Το παρακάτω αρχείο αποτελεί τον ήχο που ακούγεται στο παρασκήνιο της εφαρμογής.

Αρχεία **model,data.pickle,checkpoint**: Όλα μαζί αποτελούν το μοντέλο και τα δεδομένα που δημιουργούνται κατά την εκπαίδευση του προγράμματος.

Βιβλιοθήκες

Έκδοση Python 3.10

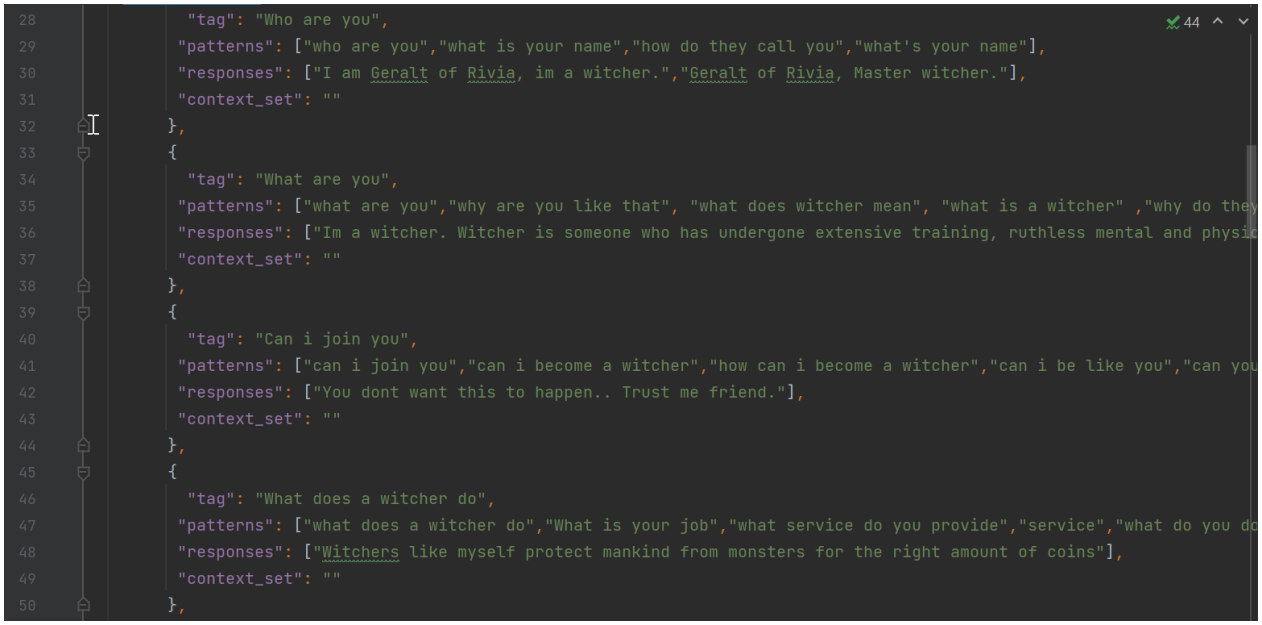
```
import nltk
from nltk.stem.lancaster import LancasterStemmer
import pygame
stemmer = LancasterStemmer()

import numpy
import tflearn
import tensorflow
import random
import json
import pickle
from tkinter import *
```

Απλώς μεταβείτε στο CMD και πληκτρολογήστε: pip install "package name". Όπου θα αντικαταστήσετε το "package_name" με όλες τις καταχωρήσεις που αναφέρονται παραπάνω.

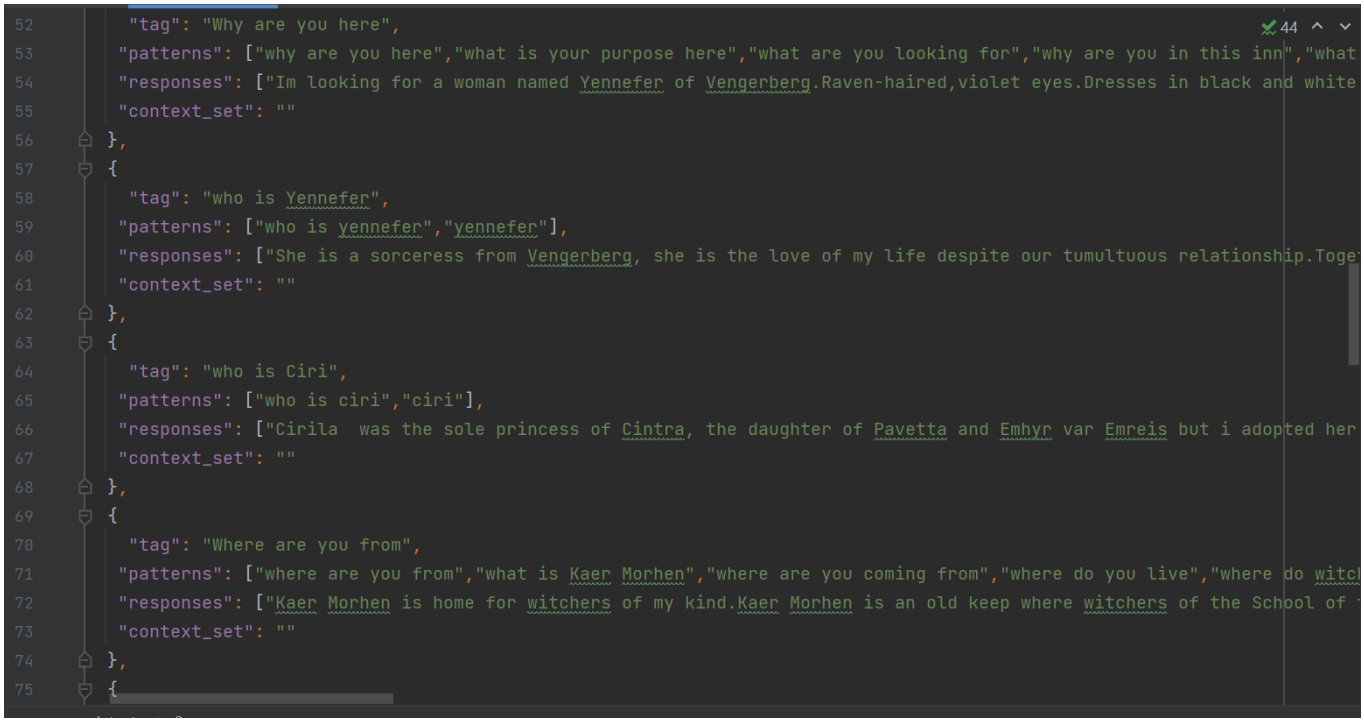
Δεδομένα

Αυτό που κάνουμε με το αρχείο JSON είναι να δημιουργούμε μια δέσμη μηνυμάτων που είναι πιθανό να πληκτρολογήσει ο χρήστης και να τα αντιστοιχίσει σε μια ομάδα κατάλληλων απαντήσεων. Το tag σε κάθε λεξικό του αρχείου υποδεικνύει την ομάδα στην οποία ανήκει και κάθε μήνυμα. Με αυτά τα δεδομένα θα εκπαιδεύσουμε ένα νευρωνικό δίκτυο να παίρνει μια πρόταση λέξεων και να την ταξινομεί ως μία από τα tags του αρχείου μας. Στη συνέχεια, μπορούμε απλώς να λάβουμε μια απάντηση από αυτές τις ομάδες και να την εμφανίσουμε στον χρήστη. Όσο περισσότερες ετικέτες, απαντήσεις και μοτίβα παρέχετε στο chatbot τόσο καλύτερο και πιο περίπλοκο θα είναι.

```
{
 "intents": [
       {"tag": "NoMatch",
        "patterns": [],
        "responses": ["Sorry,i can't understand you", "Sorry i cant answer this question", "Not sure I understand", "Ho
        "context_set": ""
       },
       {
         "tag": "welcome",
        "patterns": ["Hello there","Hi there","Hey", "Hello", "Good day","Hi", "How are you", "Is anyone there?" , "Wha
        "responses": ["Greetings", "Hello", "Hi there, how can I help?","Hey"],
        "context_set": ""
       },
       {
         "tag": "bye",
        "patterns": ["Bye","See you soon", "See you later", "Goodbye", "Nice chatting to you, bye", "Till next time"],
        "responses": ["See you!", "Farewell", "Be well."],
        "context_set": ""
       },
       {
         "tag": "thanks",
        "patterns": ["Thanks", "Thank you", "That's helpful","Awesome" ,"Awesome, thanks", "Thanks for helping me"],
        "responses": ["Happy to help!", "Any time!", "My pleasure" , "Your welcome"],
        "context_set": ""
```



```
        "tag": "Who are you",
        "patterns": ["who are you","what is your name","how do they call you","what's your name"],
        "responses": ["I am Geralt of Rivia, im a witcher.","Geralt of Rivia, Master witcher."],
        "context_set": ""
      },
      {
        "tag": "What are you",
        "patterns": ["what are you","why are you like that", "what does witcher mean", "what is a witcher" ,"why do they
        "responses": ["Im a witcher. Witcher is someone who has undergone extensive training, ruthless mental and physi
        "context_set": ""
      },
      {
        "tag": "Can i join you",
        "patterns": ["can i join you","can i become a witcher","how can i become a witcher","can i be like you","can you
        "responses": ["You dont want this to happen.. Trust me friend."],
        "context_set": ""
      },
      {
        "tag": "What does a witcher do",
        "patterns": ["what does a witcher do","What is your job","what service do you provide","service","what do you do
        "responses": ["Witchers like myself protect mankind from monsters for the right amount of coins"],
        "context_set": ""
      },
```



```
    "tag": "Why are you here",
    "patterns": ["why are you here","what is your purpose here","what are you looking for","why are you in this inn","what
    "responses": ["Im looking for a woman named Yennefer of Vengerberg.Raven-haired,violet eyes.Dresses in black and white
    "context_set": ""
  },
  {
    "tag": "who is Yennefer",
    "patterns": ["who is yennefer","yennefer"],
    "responses": ["She is a sorceress from Vengerberg, she is the love of my life despite our tumultuous relationship.Toge
    "context_set": ""
  },
  {
    "tag": "who is Ciri",
    "patterns": ["who is ciri","ciri"],
    "responses": ["Cirila  was the sole princess of Cintra, the daughter of Pavetta and Emhyr var Emreis but i adopted her
    "context_set": ""
  },
  {
    "tag": "Where are you from",
    "patterns": ["where are you from","what is Kaer Morhen","where are you coming from","where do you live","where do witch
    "responses": ["Kaer Morhen is home for witchers of my kind.Kaer Morhen is an old keep where witchers of the School of
    "context_set": ""
  },
  {
```

```
 "tag": "Why do you carry two swords",
"patterns": ["why do you carry two swords","swords","weapons","what weapons do you have","What is the points of two sw
"responses": ["Well, I have two swords. One is silver for creatures that roam the wilds, one is steel for men in their
"context_set": ""
},
{
 "tag": "do you kill humans",
"patterns": ["do you kill humans","do accept death contracts"],
"responses": ["Im a witcher not a mercenary,i kill monsters,despite that i always choose the lesser evil."],
"context_set": ""
},
{
 "tag": "how did you become a witcher",
"patterns": ["how did you become a witcher","what is witcher trials","trials"],
"responses": ["In order to become a witcher you have to pass the trials.The Trials were processes which transformed th
"context_set": ""
},
{
 "tag": "Why is your hair white",
"patterns": ["why is your hair white","why are your eyes like that","why do you look different"],
"responses": ["During the Trial,i exhibited unusual tolerance for the mutagens that grant witchers their abilities. Ac
"context_set": ""
},
```

```
 "tag": "How much do you get paid",
"patterns": ["how much do you get paid","how much does your services cost","gold","coins","money"],
"responses": ["Depends on the monster im dealing with."],
"context_set": ""
},
{
 "tag": "Witcher schools",
"patterns": ["How many schools are there","school"],
"responses": ["There are a lot of witcher schools across the world some of them are School of the Wolf,School of the C
"context_set": ""
},
{
 "tag": "how old are you",
"patterns": ["how old are you","what is your age","age"],
"responses": ["I am  104 but look 30.Witchers age differently"],
"context_set": ""
},
{
 "tag": "Fight me",
"patterns": ["do you want to fight","duel me","do you want to kill me"],
"responses": ["Sorry no mood for fighting or killing at the time."],
"context_set": ""
},
```

```
    "tag": "Whats next",
   "patterns": ["where will you go after","whats next","whats your plan"],
   "responses": ["No plan , just follow the witcher's trail"],
   "context_set": ""
  },
  {
    "tag": "Super powers",
   "patterns": ["do you have superpowers","why are you special","do you use magic","are you superhuman"],
   "responses": ["I have plenty of permanent results of mutation.Some of them are: Cat-like eyes, resistance to disease,
   "context_set": ""
  },
  {
    "tag": "Triss",
   "patterns": ["do you know triss merigold","triss"],
   "responses": ["Triss is a skilled sorceress and a beautiful woman."],
   "context_set": ""
  },
  {
    "tag": "Dandelion",
   "patterns": ["dandelion","jaskier","do you know dandelion","are you aware of dandelion"],
   "responses": ["Unfortunately dandelion is a friend of mine."],
   "context_set": ""
  }
```

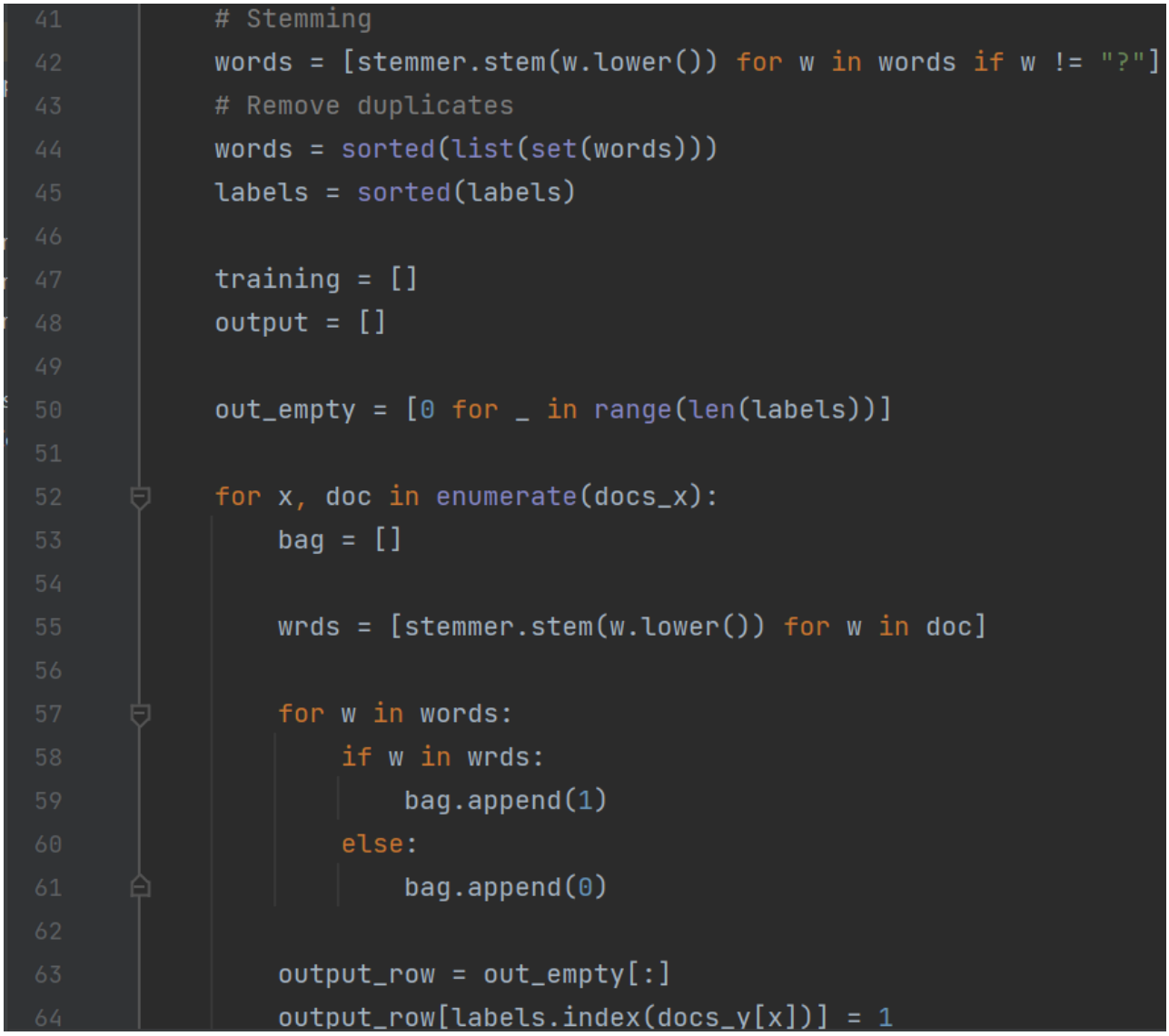
Έπειτα μέσω επαναληπτικού βρόχου εξαγάγουμε τα δεδομένα που θέλουμε. Για κάθε μοτίβο αντί να τις έχουμε ως Strings θα τις μετατρέψουμε σε μια λίστα λέξεων χρησιμοποιώντας το **nltk.word_tokenizer**. Στη συνέχεια, θα προσθέσουμε κάθε μοτίβο στη λίστα docs_x και το σχετικό tag στη λίστα docs_y.

```
try:
    with open("data.pickle", "rb") as f:
        words, labels, training, output = pickle.load(f)
except:
    # Extracting Data
    words = []
    labels = []
    docs_x = []
    docs_y = []
    '''loop through our JSON data and extract the data we want.
    For each pattern we will turn it into a list of words using nltk.word_tokenizer
    rather than having them as strings.
    We will then add each pattern into our docs_x list and its associated tag into the docs_y list'''
    for intent in data["intents"]:
        for pattern in intent["patterns"]:
            # Tokenize aka get all words in our pattern with nltk
            wrds = nltk.word_tokenize(pattern)
            words.extend(wrds)
            docs_x.append(wrds)
            docs_y.append(intent["tag"])

        if intent["tag"] not in labels:
            labels.append(intent["tag"])
```

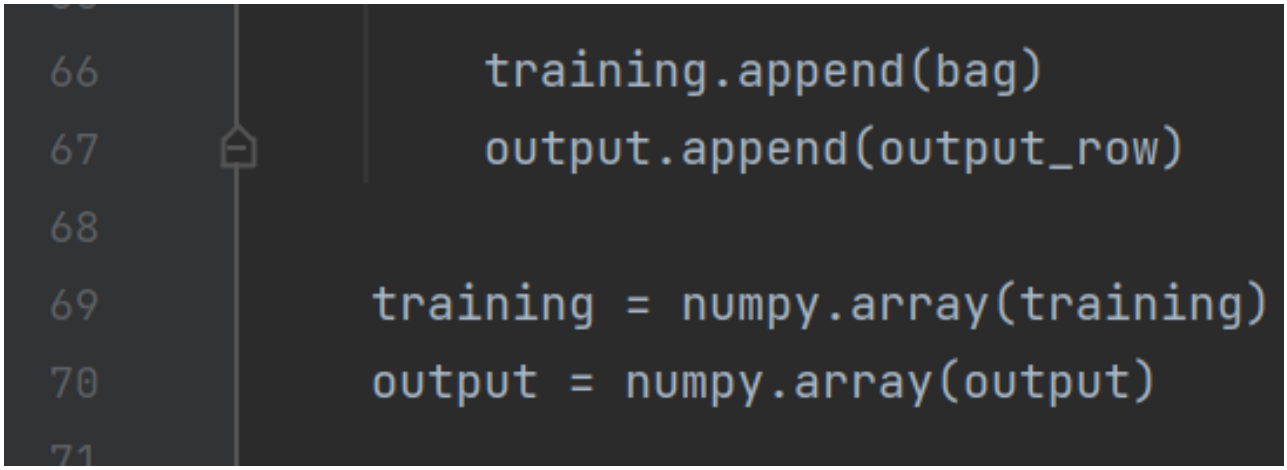
Προ επεξεργασία

Τώρα που φορτώσαμε τα δεδομένα μας και δημιουργήσαμε ένα βασικό λεξιλόγιο, μετατρέπουμε τα δεδομένα σε σωρό από λέξεις. Όπως γνωρίζουμε τα νευρωνικά δίκτυα και οι αλγόριθμοι μηχανικής μάθησης απαιτούν αριθμητική είσοδο. Έτσι, η λίστα με τα strings δεν μας κάνει. Χρειαζόμαστε κάποιον τρόπο να αναπαραστήσουμε τις προτάσεις μας με αριθμούς και εδώ μπαίνει μια ‘’τσάντα’’ ή ‘’σάκος’’ από λέξεις. Αυτό που θα κάνουμε είναι να αναπαραστήσουμε κάθε πρόταση με μια λίστα. Κάθε θέση στη λίστα θα αντιπροσωπεύει μια λέξη από το λεξιλόγιό μας. Αν η θέση στη λίστα είναι 1 τότε αυτό θα σημαίνει ότι η λέξη υπάρχει στην πρόταση μας, αν είναι 0 τότε η λέξη δεν υπάρχει. Ομοίως με τσάντες λέξεων θα δημιουργήσουμε λίστες εξόδου στο μήκος του αριθμού των tags που έχουμε στο σύνολο των δεδομένων μας. Κάθε θέση στη λίστα θα αντιπροσωπεύει ένα ξεχωριστό tag με 1 αν υπάρχει και 0 αν δεν υπάρχει.

Βιτάκης Αθανάσιος/ Π19247
Αυγερινός Χρήστος/ Π19020
Παναγιωτόπουλος Δημήτριος /Π19130



```
    # Stemming
    words = [stemmer.stem(w.lower()) for w in words if w != "?"]
    # Remove duplicates
    words = sorted(list(set(words)))
    labels = sorted(labels)

    training = []
    output = []

    out_empty = [0 for _ in range(len(labels))]

    for x, doc in enumerate(docs_x):
        bag = []

        wrds = [stemmer.stem(w.lower()) for w in doc]

        for w in words:
            if w in wrds:
                bag.append(1)
            else:
                bag.append(0)

        output_row = out_empty[:]
        output_row[labels.index(docs_y[x])] = 1
```



```
            training.append(bag)
            output.append(output_row)


    training = numpy.array(training)
    output = numpy.array(output)
```

Βιτάκης Αθανάσιος/ Π19247
Αυγερινός Χρήστος/ Π19020
Παναγιωτόπουλος Δημήτριος /Π19130

Δημιουργία Μοντέλου

Τώρα που έχουμε προ επεξεργαστεί όλα τα δεδομένα μας, πρέπει να δημιουργήσουμε και να εκπαιδεύσουμε ένα μοντέλο. Για τους σκοπούς μας θα χρησιμοποιήσουμε ένα αρκετά τυπικό νευρωνικό δίκτυο τροφοδοσίας με δύο κρυφά επίπεδα. Ο στόχος του δικτύου μας θα είναι να δούμε μια τσάντα από λέξεις και βρούμε την κατάλληλη τάξη στην οποία ανήκουν.(Ένα από τα tags του object.json)

```
tensorflow.compat.v1.reset_default_graph()

net = tflearn.input_data(shape=[None, len(training[0])])
net = tflearn.fully_connected(net, 8)
net = tflearn.fully_connected(net, 8)
net = tflearn.fully_connected(net, len(output[0]), activation="softmax")
net = tflearn.regression(net)

model = tflearn.DNN(net)
```

Αφού έχουμε ρυθμίσει το μοντέλο μας, το επόμενο στάδιο είναι να το εκπαιδεύσουμε στα δεδομένα μας. Για να το κάνουμε αυτό θα προσαρμόσουμε τα δεδομένα μας στο μοντέλο. Ο αριθμός των εποχών(epoch) που ορίζουμε είναι οι φορές που το μοντέλο θα δει τις ίδιες πληροφορίες κατά την εκπαίδευση.

Μόλις ολοκληρώσουμε την εκπαίδευση του μοντέλου, μπορούμε να το αποθηκεύσουμε στο αρχείο model.tflearn για μεταγενέστερη χρήση(πχ discord bot).

```
model.fit(training, output, n_epoch=1000, batch_size=8, show_metric=True)
model.save("model.tflearn")
```

Βιτάκης Αθανάσιος/ Π19247
Αυγερινός Χρήστος/ Π19020
Παναγιωτόπουλος Δημήτριος /Π19130

Διαδικασία πρόβλεψης

Πρέπει να θυμόμαστε ότι το μοντέλο μας δεν δέχεται είσοδο από strings , χρειάζεται μια τσάντα από λέξεις. Πρέπει επίσης να συνειδητοποιήσουμε ότι το μοντέλο μας δεν παράγει προτάσεις αλλά δημιουργεί μια λίστα πιθανοτήτων για όλες τις τάξεις μας(tags).

Η συνάρτηση **bag_of_words()** θα μετατρέψει τη συμβολοσειρά μας σε μια τσάντα λέξεων χρησιμοποιώντας τη λίστα λέξεων που δημιουργήσαμε. Η συνάρτηση **chat()** θα χειριστεί τη λήψη μιας πρόβλεψης από το μοντέλο και τη λήψη μιας κατάλληλης απάντησης από το αρχείο απαντήσεων JSON.

```
89  def bag_of_words(s, words):
90      bag = [0 for _ in range(len(words))]
91
92      s_words = nltk.word_tokenize(s)
93      s_words = [stemmer.stem(word.lower()) for word in s_words]
94
95      for se in s_words:
96          for i, w in enumerate(words):
97              if w == se:
98                  bag[i] = 1
99
100     return numpy.array(bag)
101
102
```

```
103 def chat(input):
104     results = model.predict([bag_of_words(input, words)])[0]
105     results_index = numpy.argmax(results)
106     tag = labels[results_index]
107
108     if results[results_index] > 0.75:
109         for tg in data["intents"]:
110             if tg['tag'] == tag:
111                 responses = tg['responses']
112
113         msg = random.choice(responses)
114     else:
115         msg = random.choice(data['intents'][0]['responses'])
116
117     return msg
```

Βιτάκης Αθανάσιος/ Π19247
Αυγερινός Χρήστος/ Π19020
Παναγιωτόπουλος Δημήτριος /Π19130

Γραφικό περιβάλλον(GUI)

```
# Color,fonts etc
BG_GRAY = "#ABB2B9"
BG_COLOR = "#17202A"
TEXT_COLOR = "#EAECEE"
FONT = "Garamond 14"
FONT_BOLD = "Garamond 13 bold"


# GUI
class ChatApplication:
    def __init__(self):
        self.window = Tk()
        self._setup_main_window()
        self.init_message("Narrator", "You are an ordinary peasant who live in the city of Novigrand"
                                      " and decided to spend your afternoon in your local inn. "
                                      "There you notice a white haired man with two swords on his back that "
                                      "seems highly unusual. "
                                      "You asked the innkeeper about it and he replied that he is a witcher. "
                                      " Not knowing what this mean , you decide to start a conversation with stranger. ")
        self.init_message("Narrator", "Say hi to begin.")

```

```
    def run(self):
        self.window.mainloop()

    def _setup_main_window(self):
        self.window.title("Conversation")
        self.window.resizable(width=True, height=True)
        self.window.configure(width=550, height=550, bg=BG_COLOR)
        # head label
        head_label = Label(self.window, bg=BG_COLOR, fg=TEXT_COLOR,
                           text="Speak...", font=FONT_BOLD, pady=10)
        head_label.place(relwidth=1)

        # tiny divider
        line = Label(self.window, width=450, bg=BG_GRAY)
        line.place(relwidth=1, rely=0.07, relheight=0.012)

        # text widget
        self.text_widget = Text(self.window, width=20, height=2, bg=BG_COLOR, fg=TEXT_COLOR, font=FONT, padx=5, pady=5)
        self.text_widget.place(relheight=0.745, relwidth=1, rely=0.08)
        self.text_widget.configure(cursor="arrow", state=DISABLED)

        # bottom label
        bottom_label = Label(self.window, bg=BG_GRAY, height=80)
```

```
        # message entry box
        self.msg_entry = Entry(bottom_label, bg="#2C3E50", fg=TEXT_COLOR, font=FONT)
        self.msg_entry.place(relwidth=0.74, relheight=0.06, rely=0.008, relx=0.011)
        self.msg_entry.focus()
        self.msg_entry.bind("<Return>", self._on_enter_pressed)

        # say button
        say_button = Button(bottom_label, text="Say", font=FONT_BOLD, width=20, bg=BG_GRAY,
                            command=lambda: self._on_enter_pressed(None))
        say_button.place(relx=0.77, rely=0.008, relheight=0.06, relwidth=0.22)

```

Βιτάκης Αθανάσιος/ Π19247
Αυγερινός Χρήστος/ Π19020
Παναγιωτόπουλος Δημήτριος /Π19130

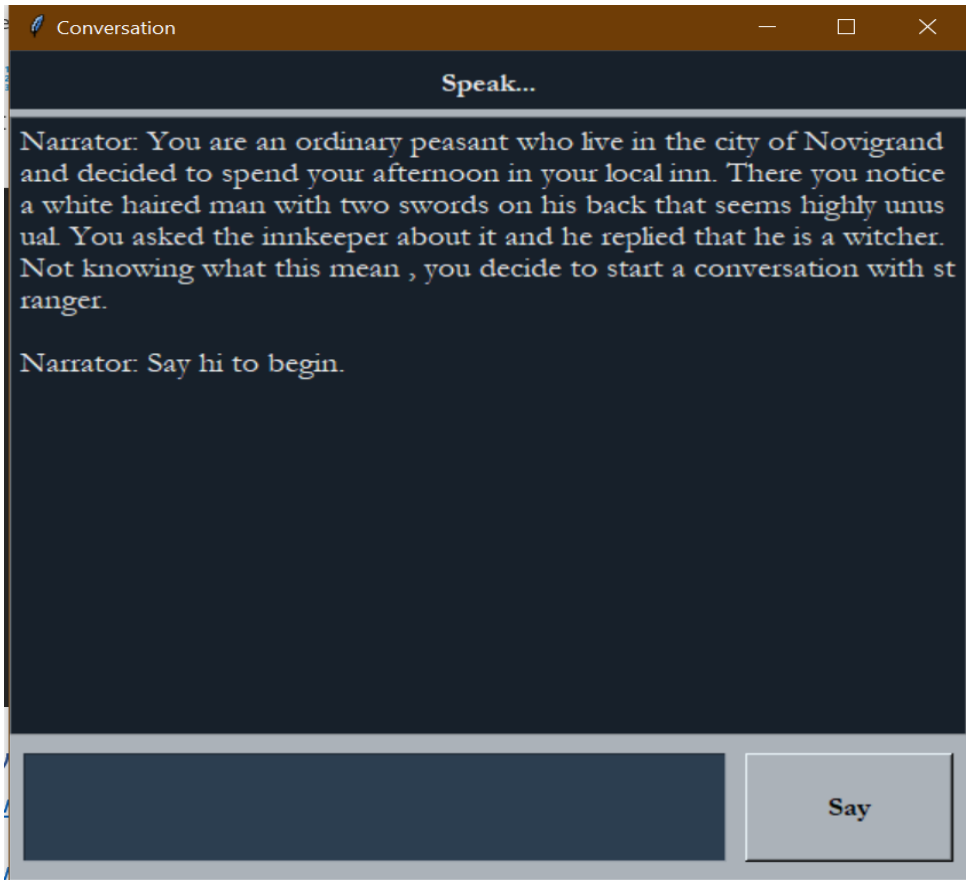
```
177     def _on_enter_pressed(self, event):
178         msg = self.msg_entry.get()
179         self._insert_message(msg, "You")
180 
181     def _insert_message(self, msg, sender):
182         if not msg:
183             return
184         self.msg_entry.delete(0, END)
185         msg1 = f"{sender}: {msg}\n\n"
186         self.text_widget.configure(state=NORMAL)
187         self.text_widget.insert(END, msg1)
188         self.text_widget.configure(state=DISABLED)
189 
190         msg2 = f"{'Witcher'}: {chat(msg)}\n\n"
191         self.text_widget.configure(state=NORMAL)
192         self.text_widget.insert(END, msg2)
193         self.text_widget.configure(state=DISABLED)
194 
195         self.text_widget.see(END)
```

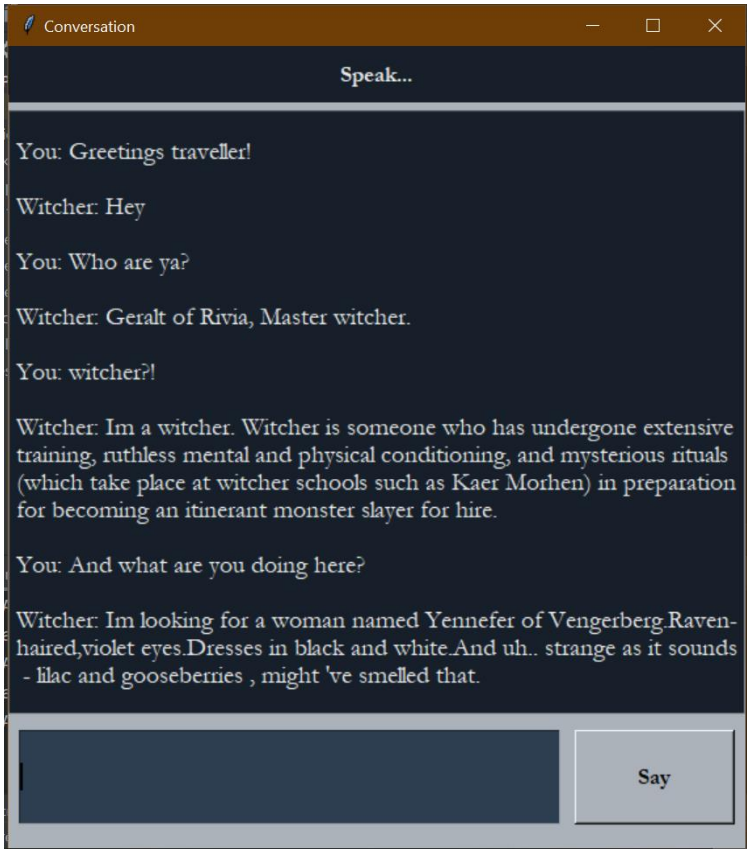
```
197     def init_message(self, talker, msg):
198         msg2 = f"{talker}: {msg}\n\n"
199         self.text_widget.configure(state=NORMAL)
200         self.text_widget.insert(END, msg2)
201         self.text_widget.configure(state=DISABLED)
202 
203 
204 # Music set up
205 pygame.mixer.init()
206 crash_sound = pygame.mixer.Sound("ambience.mp3")
207 crash_sound.set_volume(0.2)
208 crash_sound.play(loops=-1)
209 
210 # Start Application
211 app = ChatApplication()
212 app.run()
213 
```

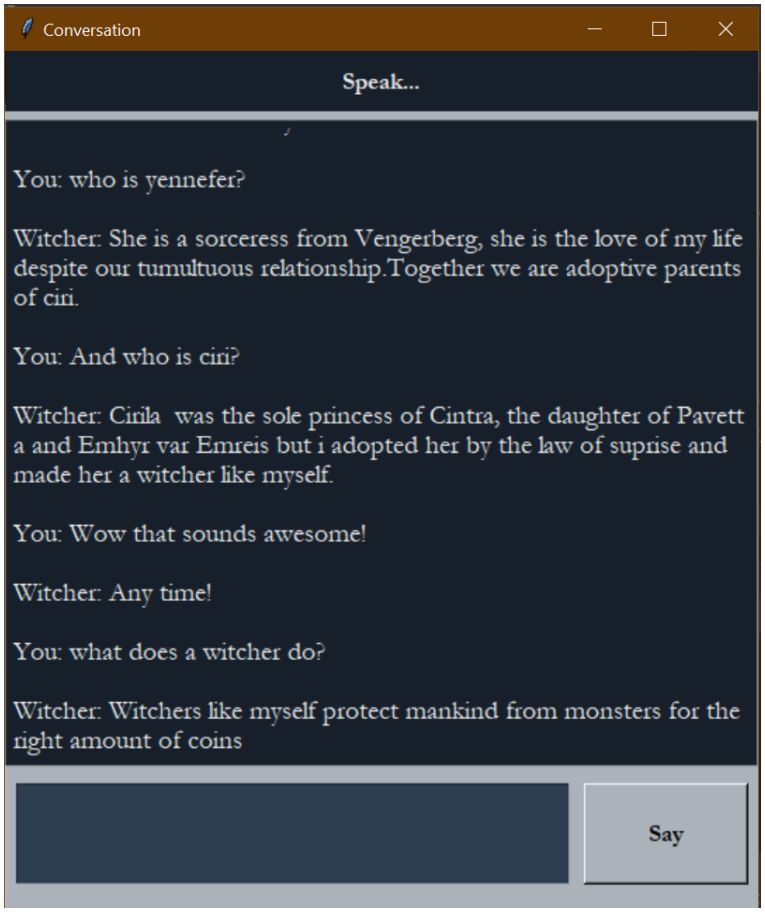
Παραδείγματα εκτέλεσης

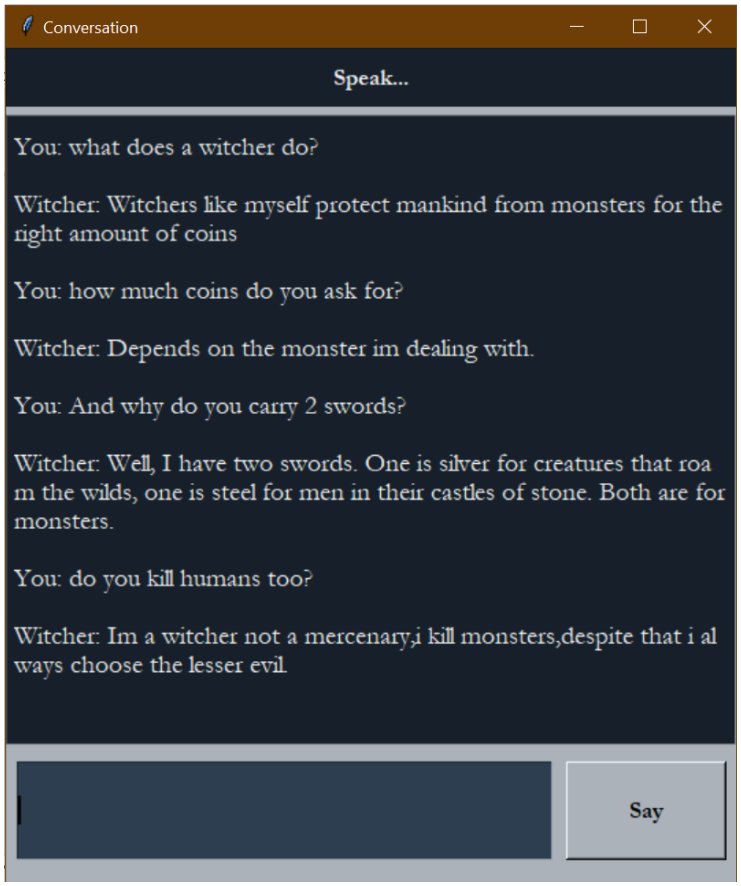
Κατά την εκτέλεση του προγράμματος θα εμφανιστεί το παράθυρο της εφαρμογής το οποίο θα περιλαμβάνει και μουσική στο παρασκηνιο.

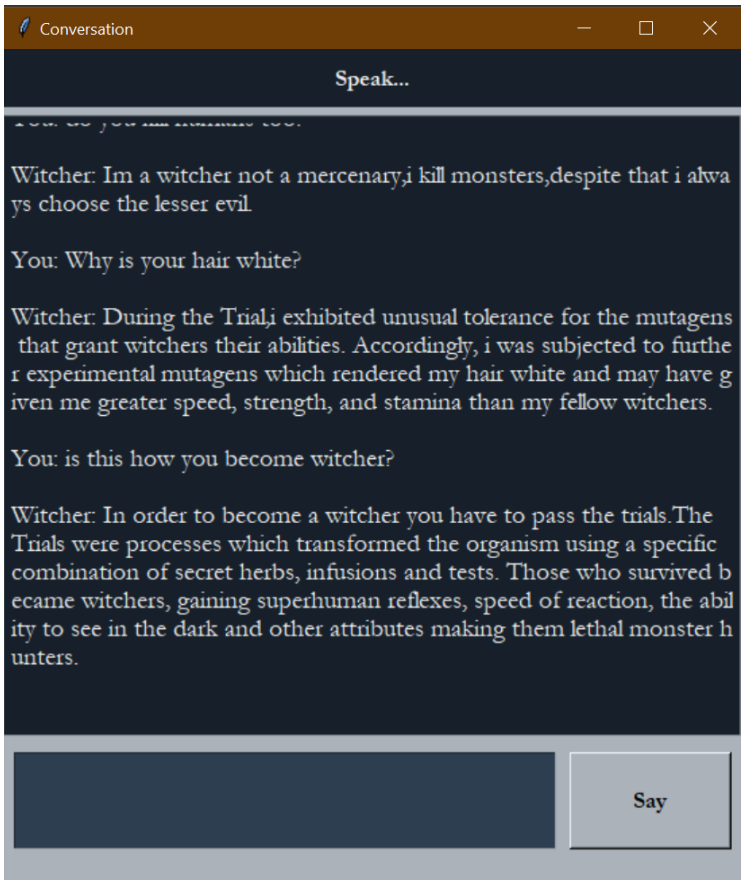
```
Training Step: 11991  | total loss: 0.00033 | time: 0.003s
| Adam | epoch: 1000 | loss: 0.00033 - acc: 1.0000 -- iter: 24/96
Training Step: 11992  | total loss: 0.00031 | time: 0.004s
| Adam | epoch: 1000 | loss: 0.00031 - acc: 1.0000 -- iter: 32/96
Training Step: 11993  | total loss: 0.00033 | time: 0.005s
| Adam | epoch: 1000 | loss: 0.00033 - acc: 1.0000 -- iter: 40/96
Training Step: 11994  | total loss: 0.00034 | time: 0.006s
| Adam | epoch: 1000 | loss: 0.00034 - acc: 1.0000 -- iter: 48/96
Training Step: 11995  | total loss: 0.00032 | time: 0.007s
| Adam | epoch: 1000 | loss: 0.00032 - acc: 1.0000 -- iter: 56/96
Training Step: 11996  | total loss: 0.00030 | time: 0.008s
| Adam | epoch: 1000 | loss: 0.00030 - acc: 1.0000 -- iter: 64/96
Training Step: 11997  | total loss: 0.00030 | time: 0.009s
| Adam | epoch: 1000 | loss: 0.00030 - acc: 1.0000 -- iter: 72/96
Training Step: 11998  | total loss: 0.00028 | time: 0.010s
| Adam | epoch: 1000 | loss: 0.00028 - acc: 1.0000 -- iter: 80/96
Training Step: 11999  | total loss: 0.00027 | time: 0.011s
| Adam | epoch: 1000 | loss: 0.00027 - acc: 1.0000 -- iter: 88/96
Training Step: 12000  | total loss: 0.00027 | time: 0.012s
| Adam | epoch: 1000 | loss: 0.00027 - acc: 1.0000 -- iter: 96/96
--
```

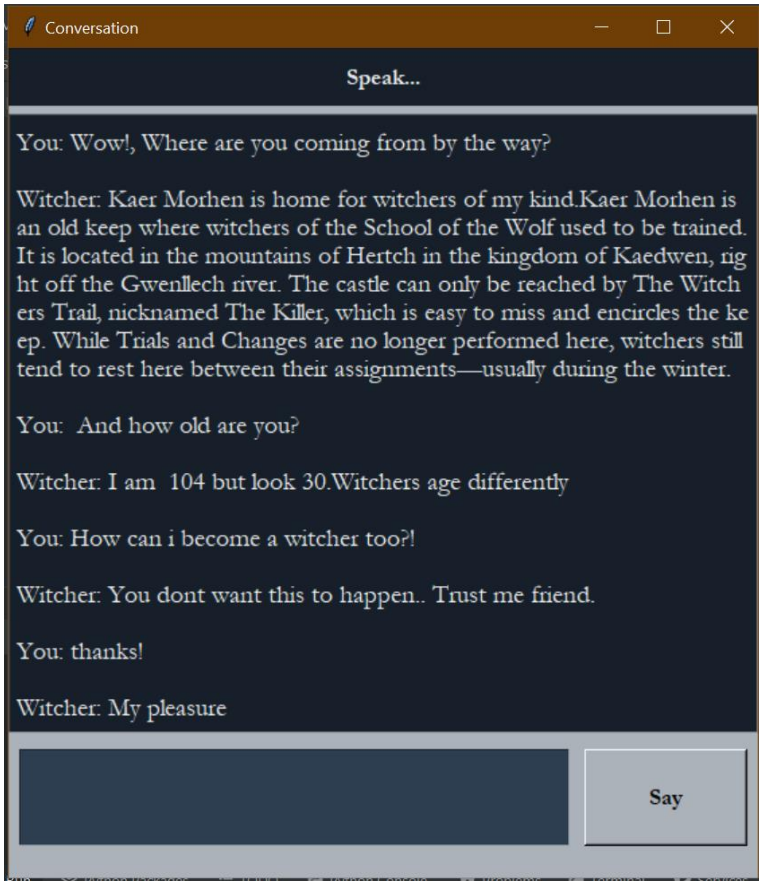


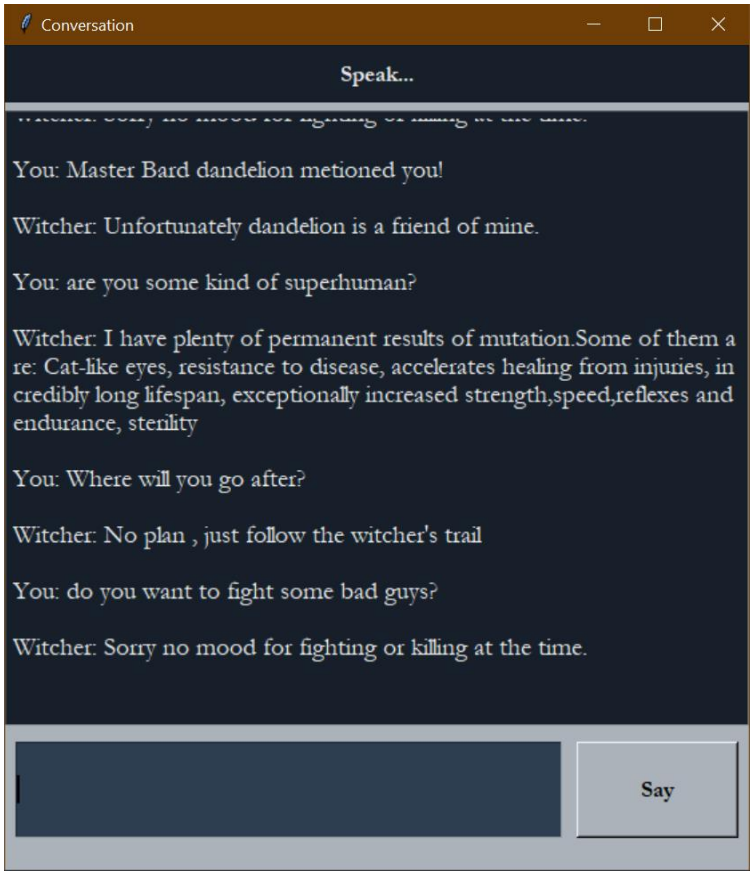

Conversation
Speak...
You: Greetings traveller!
Witcher: Hey
You: Who are ya?
Witcher: Geralt of Rivia, Master witcher.
You: witcher?!
Witcher: Im a witcher. Witcher is someone who has undergone extensive training, ruthless mental and physical conditioning, and mysterious rituals (which take place at witcher schools such as Kaer Morhen) in preparation for becoming an itinerant monster slayer for hire.
You: And what are you doing here?
Witcher: Im looking for a woman named Yennefer of Vengerberg.Raven-haired,violet eyes.Dresses in black and white.And uh.. strange as it sounds - lilac and gooseberries , might 've smelled that.
Say

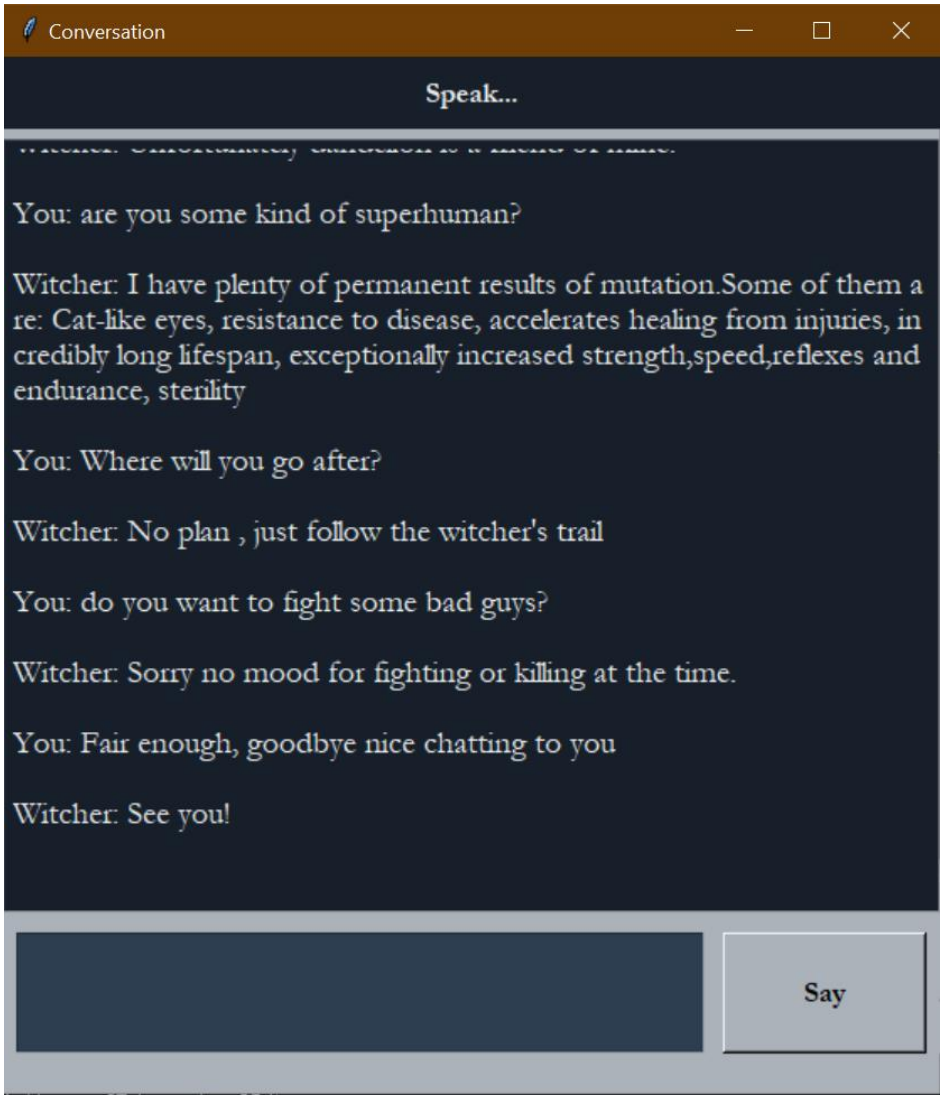

Conversation
Speak...
You: who is yennefer?
Witcher: She is a sorceress from Vengerberg, she is the love of my life despite our tumultuous relationship.Together we are adoptive parents of ciri.
You: And who is ciri?
Witcher: Cirila was the sole princess of Cintra, the daughter of Pavett a and Emhyr var Emreis but i adopted her by the law of suprise and made her a witcher like myself.
You: Wow that sounds awesome!
Witcher: Any time!
You: what does a witcher do?
Witcher: Witchers like myself protect mankind from monsters for the right amount of coins
Say


Conversation
Speak...
You: what does a witcher do?
Witcher: Witchers like myself protect mankind from monsters for the right amount of coins
You: how much coins do you ask for?
Witcher: Depends on the monster im dealing with.
You: And why do you carry 2 swords?
Witcher: Well, I have two swords. One is silver for creatures that roa m the wilds, one is steel for men in their castles of stone. Both are for monsters.
You: do you kill humans too?
Witcher: Im a witcher not a mercenary,i kill monsters,despite that i al ways choose the lesser evil.
Say


Conversation
Speak...
Witcher: Im a witcher not a mercenary,i kill monsters,despite that i alwa ys choose the lesser evil.
You: Why is your hair white?
Witcher: During the Trial,i exhibited unusual tolerance for the mutagens that grant witchers their abilities. Accordingly, i was subjected to furthe r experimental mutagens which rendered my hair white and may have g iven me greater speed, strength, and stamina than my fellow witchers.
You: is this how you become witcher?
Witcher: In order to become a witcher you have to pass the trials.The Trials were processes which transformed the organism using a specific combination of secret herbs, infusions and tests. Those who survived b ecame witchers, gaining superhuman reflexes, speed of reaction, the abil ity to see in the dark and other attributes making them lethal monster h unters.
Say


Conversation
Speak...
You: Wow!, Where are you coming from by the way?
Witcher: Kaer Morhen is home for witchers of my kind.Kaer Morhen is an old keep where witchers of the School of the Wolf used to be trained. It is located in the mountains of Hertch in the kingdom of Kaedwen, rig ht off the Gwenllech river. The castle can only be reached by The Witch ers Trail, nicknamed The Killer, which is easy to miss and encircles the ke ep. While Trials and Changes are no longer performed here, witchers still tend to rest here between their assignments—usually during the winter.
You: And how old are you?
Witcher: I am 104 but look 30.Witchers age differently
You: How can i become a witcher too?!
Witcher: You dont want this to happen.. Trust me friend.
You: thanks!
Witcher: My pleasure
Say

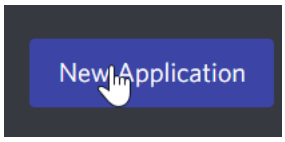

Conversation
Speak...
You: Master Bard dandelion metioned you!
Witcher: Unfortunately dandelion is a friend of mine.
You: are you some kind of superhuman?
Witcher: I have plenty of permanent results of mutation.Some of them a re: Cat-like eyes, resistance to disease, accelerates healing from injuries, in credibly long lifespan, exceptionally increased strength,speed,reflexes and endurance, sterility
You: Where will you go after?
Witcher: No plan , just follow the witcher's trail
You: do you want to fight some bad guys?
Witcher: Sorry no mood for fighting or killing at the time.
Say


Conversation
Speak...
You: are you some kind of superhuman?
Witcher: I have plenty of permanent results of mutation.Some of them a re: Cat-like eyes, resistance to disease, accelerates healing from injuries, in credibly long lifespan, exceptionally increased strength,speed,reflexes and endurance, sterility
You: Where will you go after?
Witcher: No plan , just follow the witcher's trail
You: do you want to fight some bad guys?
Witcher: Sorry no mood for fighting or killing at the time.
You: Fair enough, goodbye nice chatting to you
Witcher: See you!
Say

Σύνδεση με discord

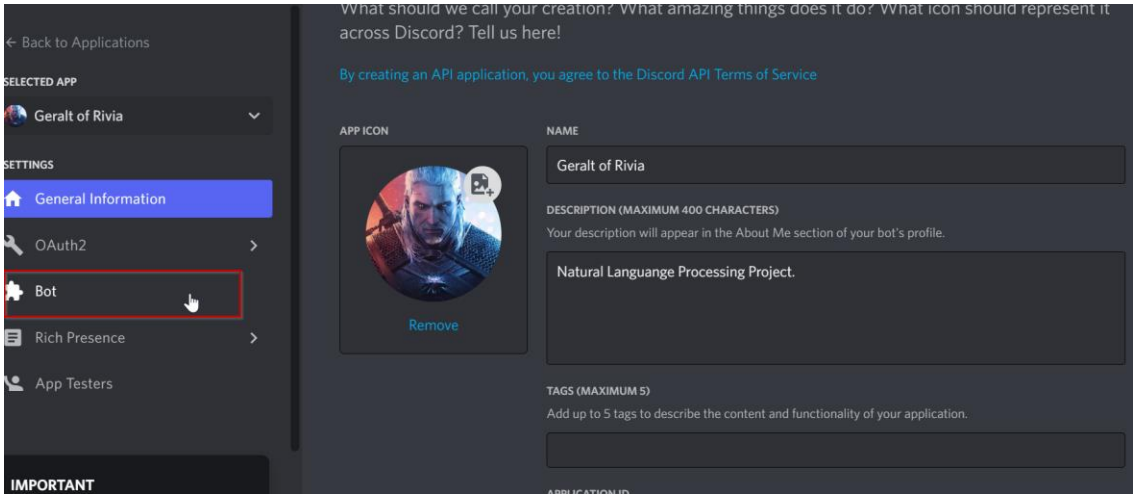
Εφόσον έχετε λογαριασμό discord και δικό σας discord server.Μεταβείτε στο https://discord.com/developers/applications

1. Επιλέξτε **New Application** και γράψτε το όνομα που επιθυμείτε.

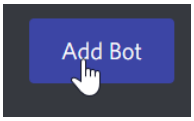


Μπορείτε να βάλετε περιγραφή App icon κλπ.

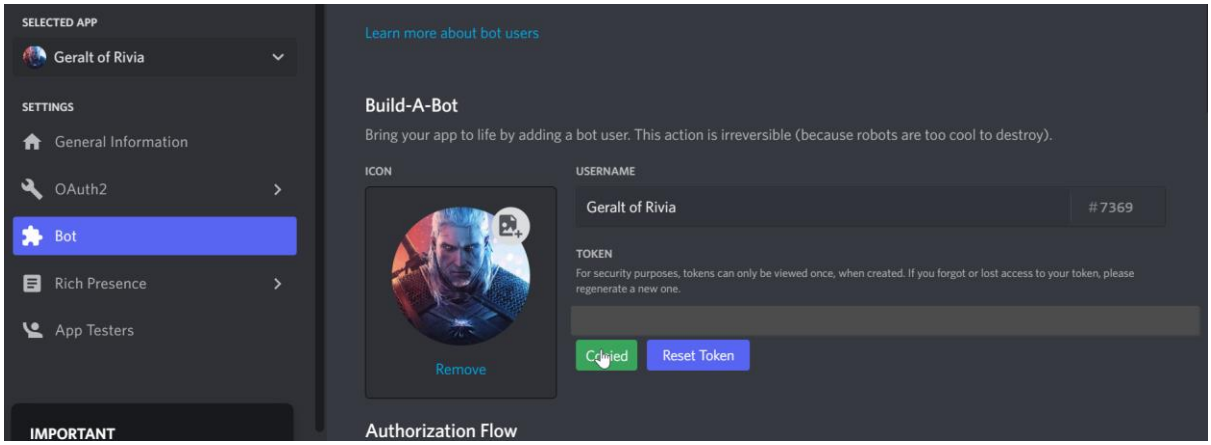
2. Πατήστε στην **ρύθμιση Bot.**



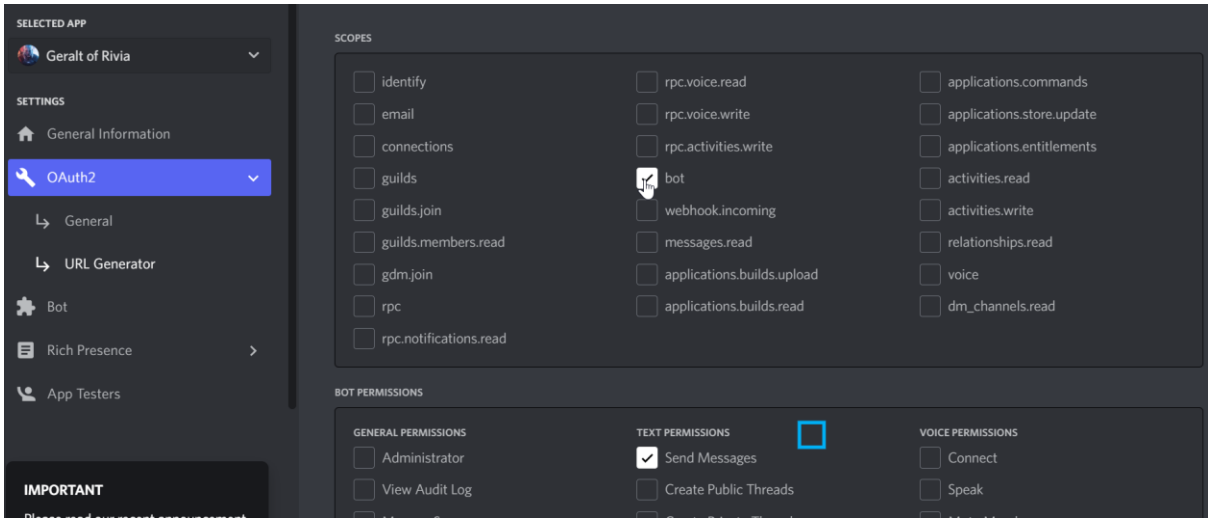
3. **Add Bot**.



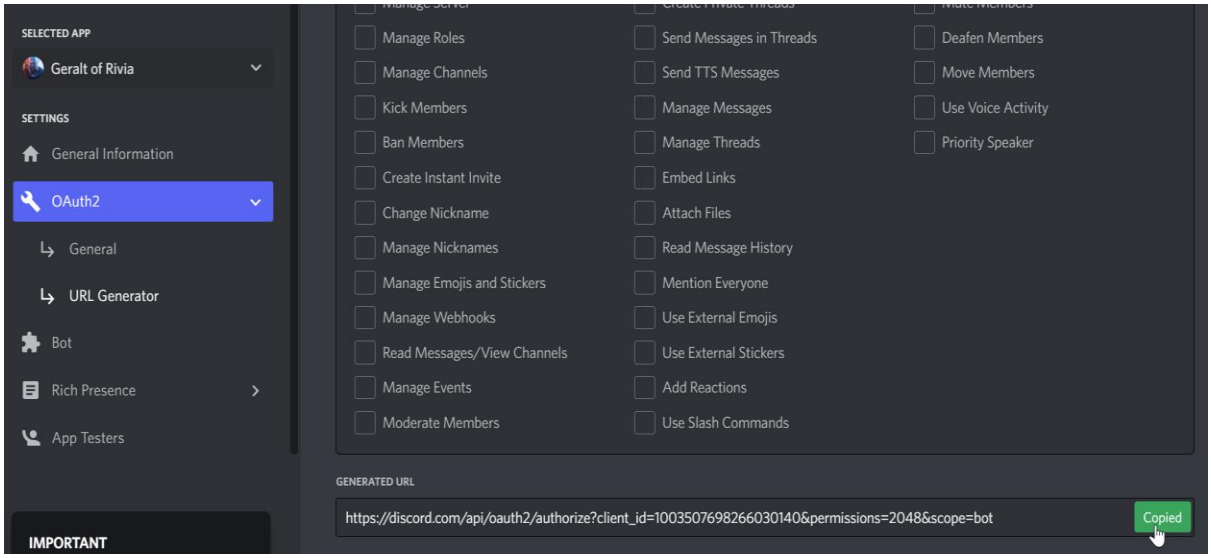
4. Αντιγράψτε το Token και αλλάξτε την γραμμή **client.run('Token')** στο αρχείο **discordConnect.py** με το Token που αντιγράψατε.



5. Πατήστε στην ρύθμιση **OAuth2 -> URL Generator** και επιλέξτε το Scope **bot** και **τα δικαιώματα** που επιθυμείτε το bot σας να έχει.

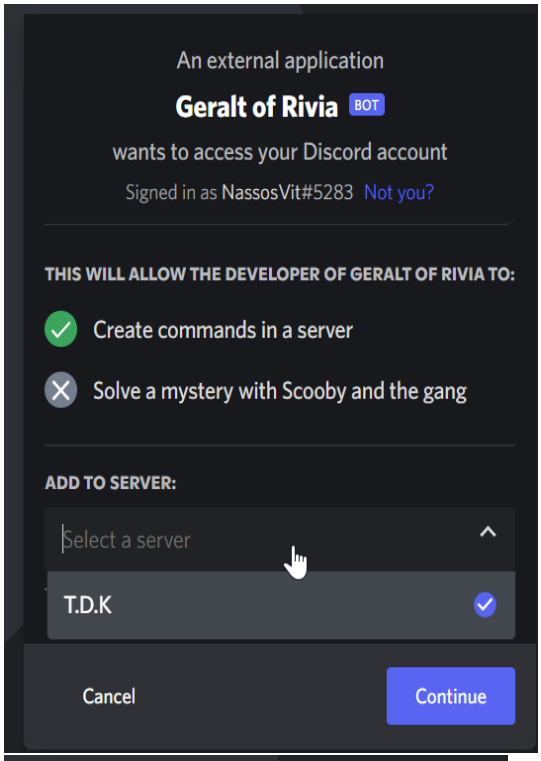


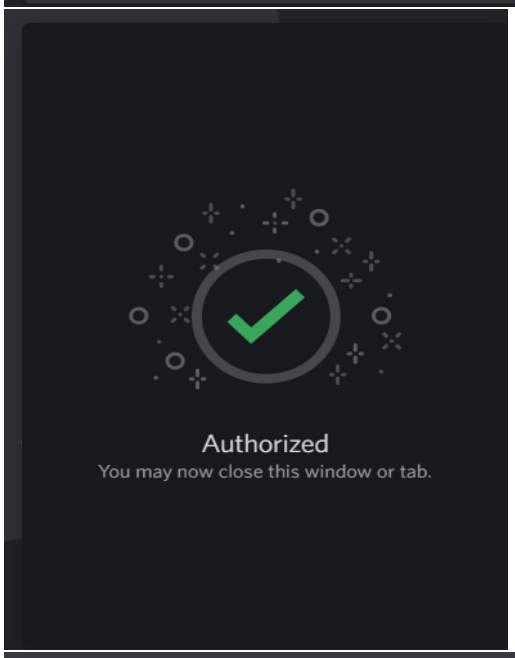
6. Αντιγράψτε το **link** που παράγεται



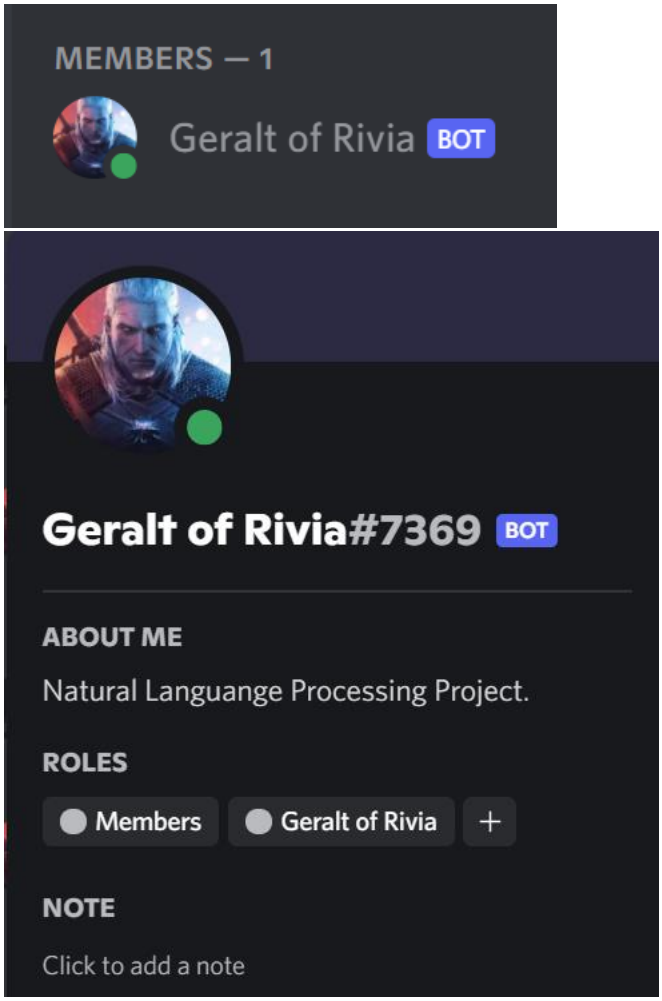
Βιτάκης Αθανάσιος/ Π19247
Αυγερινός Χρήστος/ Π19020
Παναγιωτόπουλος Δημήτριος /Π19130

7. Επικολλήστε το link που αντιγράψατε στον browser σας και επιλέξτε τον server που επιθυμείτε.



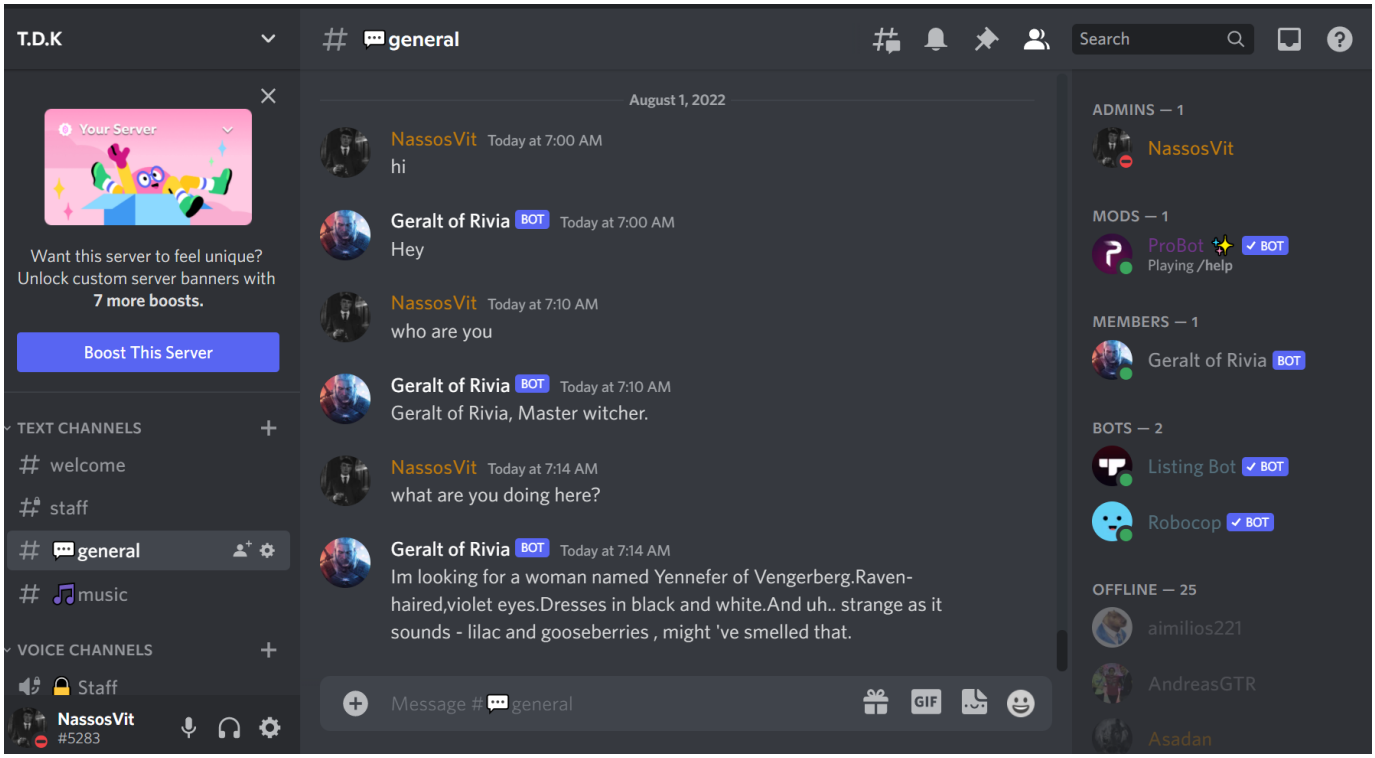


8. Εκτελέστε το αρχείο **discordConnect.py**



Το ChatBot θα είναι ενεργό όσο τρέχει και το πρόγραμμα ***discordConnect.py***

Βιτάκης Αθανάσιος/ Π19247
Αυγερινός Χρήστος/ Π19020
Παναγιωτόπουλος Δημήτριος /Π19130



Βιβλιογραφικές πηγές

- https://www.telusinternational.com/articles/the-future-of-chatbots
- https://www.xenonstack.com/blog/difference-between-nlp-nlu-nlg
- https://www.investopedia.com/terms/c/chatbot.asp
- https://www.oracle.com/chatbots/what-is-a-chatbot/
- https://docs.python.org/3/library/json.html
- https://towardsdatascience.com/how-to-build-your-own-ai-chatbot-on-discord-c6b3468189f4