

CAPSTONE DOCUMENT

The Model Answer

Hiba Talat

# What is “The Model Answer”

This project aims to use “Natural Language Processing” to clean the data and compare the result with the already existing data from Mongo DB. This system used number of libraries which I tried my best to explain in the code and also put next to the code related to it for easy understanding of the user.

# What is Natural language processing (NLP)

NLP is about developing applications that can understand human languages. Some practical examples of NLP are speech recognition for example google voice search, Alexa, Siri, sentiment analysis or understanding the text etc.

Along the computer evolution came the inspiring research on making the computer know natural human language and interact with the humans i.e., applying natural language processing to normal computer practice and beyond.

Natural language processing can be stated as a theoretical approach surrounded by analysis and manipulation of natural language texts frequently spoken by humans. This can be performed at many levels of linguistic analysis to achieve a ‘human-like’ approach in processing of tasks and other problems.

NLP defined standard system but along with a group of numerous language processing techniques and methods. Moreover, in view of enabling the user and standing true to the name, texts must be of natural language practice and not a set of selected texts that could be used for processing. Because the later approach would certainly miss out the real meaning of natural language processing, where semantics plays a role, and a lot of work is going on these days in this field.

In any NLP system, many levels of linguistic analysis of the text are accomplished. This is due to humans usually breakup texts into different levels and then process or understand the language. Human-like processing in the NLP systems are considered as an essential part of artificial intelligence. The applications of NLP are flexible and are currently being researched and implemented in fields like military science, security systems, virtual reality simulation, medicine and regular computer science and artificial intelligence.

# Demonstration

“The Model Answer” is a simple representation of an interface that selects random sentence from the database (MongoDB Atlas) and display it on the GUI . Then the user will enter a sentence and in return this system will compare the two sentences after applying NLTK (Natural language tool kit ) to clean the sentences from both sources and compare.

This apparently look quite easy and a small task but is basically a base for many big databases. For example, the model answer can be developed into an exam software system. Especially, during COVID times from schools to universities and offices, online assessment is a huge thing now. Systems are automated to evaluate the student’s performance and can also use for evaluating quick small assignments because the result will be displayed in few seconds.

These evaluation systems mainly include:

* Assessments
* Automatic grading
* Online tests
* Question libraries

Beside the educational purposes, the same system can be used in the games.  In video games, the communication consists of linguistic information that is passed either via spoken content or written content. This can be translated further compare with game instructions and return the best match of available options. These features increase the realism of the games.

# System Requirements:

1. Mongodb Atlas access to collection and documents

<https://cloud.mongodb.com/v2/6121b9107a12b36bfc551739#metrics/replicaSet/612313cda2e2d25be365b868/explorer/sentences/random_sentence/find>

1. Run jupyter notebook

* [Install](https://jupyter.org/install) jupyter notebook

|  |
| --- |
| **Note:** |
| * Google colab and mybinder are not compatible with tkinter * You can use annaconda software to run the jupyter notebook * This tutorial on [YouTube](https://www.youtube.com/watch?v=AuTkAWEa06E&ab_channel=HongLy) might help you |

1. Code is available on github: Download the file name [Capstone\_final\_v1.2.ipynb](https://github.com/HibaTalat/project_final)

## User Manual

Follow the steps below to run the code in jupyter notebook:

1. Open the file in jupyter notebook.
2. Run the first cell by the heading name “Libraries” these are basic initial libraries to install.

|  |
| --- |
| **Note:** |
| The word run means to select the cell and click on run button as shown below. |

1. Setup Mongodb Atlas:
   * Install pymongodb
   * Install textblob for nltk processing
   * Run Mongodb cell to create a mongodb client and connect it with the database. This will display the number of sentences in the database.
2. Run the “randomly picked sentence” which as the name suggest will pick a random sentence from the Database.
3. To run the tkinter block of code, follow the steps below
   * Comment out “#Convert to lower case the user input” till “# lbl1.pack()”
   * Click “Enter” on the new window appears “TextBox Input”
   * Close this current window
   * Go back to jupyter notebook and uncomment the previously commented section
   * Run again this whole block on jupyter notebook
   * A new “TextBox Input” GUI will be open now enter your sentence in comparison with the one from Mongodb Atlas , click “Enter”and close the window
   * On jupyter notebook run the whole block again

|  |
| --- |
| **Note:** |
| Tkinter call for “inp” global variable has some issue so above process make sure that PrintInput() runs properly and this global variable can be called outside the function.  This library i.e., Tkinter needs to seed (initiate) the global variable first before functioning completely. |

1. Run the next block that is “#Synonyms , Lemmatize and Stemming” , this will replace with synonyms , lemmatize and stemmatize the sentence. Return the complete sentence including different variations of the sentence to compare with the one from database.

|  |
| --- |
| **Note:** |
| In case , the code doesnot run close the ” TextBox Input” window, if it is open in your background. |

1. Run the following two blocks “# Comparing two sentences and returning the percentage”. This will open a new tkinter window showing if the result is either “PASS” or “FAIL”

# Things can be improved in future

## Different GUI library:

I would like to use PyQt library as it is a modern looking GUI and comes with many powerful widgets. Where as Tkinter is an old library and has basic features. It needs a lot of time for understanding all the details of PyQt.

## Fuzzy function

Fuzzy is a python package that checks if two strings sound the same. It is very useful for catching and correcting the typos.

# Troubleshooting

Libraries you might need if your environment gives error:

* !pip install dnspython

In case dns module doesn’t found

* !pip install pymongo[srv]

!pip install pymongo[tls]

In case error in pulling the data from mongodb

* import nltk

nltk.download('wordnet')

In case the system require explicit downloading of wordnet

## References

*check match percentage between two strings python Code Example*. (n.d.). Check Match. Retrieved September 16, 2021, from https://www.codegrepper.com/code-examples/python/check+match+percentage+between+two+strings+python

GeeksforGeeks. (2020, December 11). *Python GUI - PyQt VS TKinter*. https://www.geeksforgeeks.org/python-gui-pyqt-vs-tkinter/

*Get Started with Atlas — MongoDB Atlas*. (n.d.). Mongodb Atlas. Retrieved September 16, 2021, from https://docs.atlas.mongodb.com/getting-started/

*The Python Tutorial — Python 3.9.7 documentation*. (n.d.). Python. Retrieved September 16, 2021, from https://docs.python.org/3/tutorial/

*tkinter — Python interface to Tcl/Tk — Python 3.9.7 documentation*. (n.d.). Tkinter Python. Retrieved September 16, 2021, from https://docs.python.org/3/library/tkinter.html