

Chetan Sharan

Data Scientist|Engineer|Private Pilot

IIT KanPur

An Artificial Intelligence Model for Natural Language Processing |A Case Study

# Table Of Content

[1.0 Understanding Problem Statement 2](#_Toc173254264)

[2.0 Literature Survey 3](#_Toc173254265)

[3.0 Methodology 3](#_Toc173254266)

[3.1 Preparing Data 3](#_Toc173254268)

[3.2 Setting up the Environment 4](#_Toc173254269)

[3.3 Model Selection & Fine Tuning 5](#_Toc173254270)

[3.4 Evaluation and Testing 5](#_Toc173254272)

[4.0 Conclusion 5](#_Toc173254273)

# Understanding Problem Statement

***Problem Statement:*** *Develop a state-of-the-art question-answering model leveraging the Quora Question Answer Dataset. The objective is to create an AI system capable of understanding and generating accurate responses to a variety of user queries, mimicking a human-like interaction.*

The Date set is from Quora, it contains around 56 thousand questions answers arranged in two columns. One row for each question & answer. This is mostly text data.

# Literature Survey

Before setting up the methodology of the project, it was necessary to do some literature survey. I started with ***Bag of words*** through ***RNN*** finally to ***Transformers.*** Followed by popular example on NLP that are available.

Closed book, also known as open-domain question answering, refers to the task of answering a question from memory, without being given any background information. This task is different from the usual question answering task as here context is not provided and the response has to be generated.

# Methodology

## 

Figure 1: The Methodology.

## 3.1 Preparing Data

**Accessing Data**: The data is hosted at hugging face & its easily accessible using Pandas or Datasets (hugging face) libraries.

from datasets import load\_dataset

ds = load\_dataset("toughdata/quora-question-answer-dataset")

Since this is a decently sized text data, a methodology is following step by step to preprocess the data for NLP models. The process is shown in the following figure.

Figure 2: Data Preparation.

**Data Cleaning**: Remove any unwanted text, handle inconsistencies, and ensure the data is in a clean, usable format. Regular Expressions Library is used to make the data clean.

**Tokenization**: Convert the text data into tokens that the model can understand.

**Stop Word Removal:** This step is left on purpose as I have selected high processing GPU & CPU from AWS server.

**Lemmatization:** For this project, lemmatization has been selected over stemming as we are dealing with text data of mostly English language. Also, I have procured significant GPU capacity from AWS.

## 3.2 Setting up the Environment

**Hardware:** Since I have a Dell G series laptop with basic GPU configuration, I could not use this machine for NLP process. I have selected AWS as the cloud service because of its good customer support & familiarity with the system. One Instance of ***p3.2xlarge*** with Tesla V100- 16 GiB VRAM was used as shown in fig. 3.

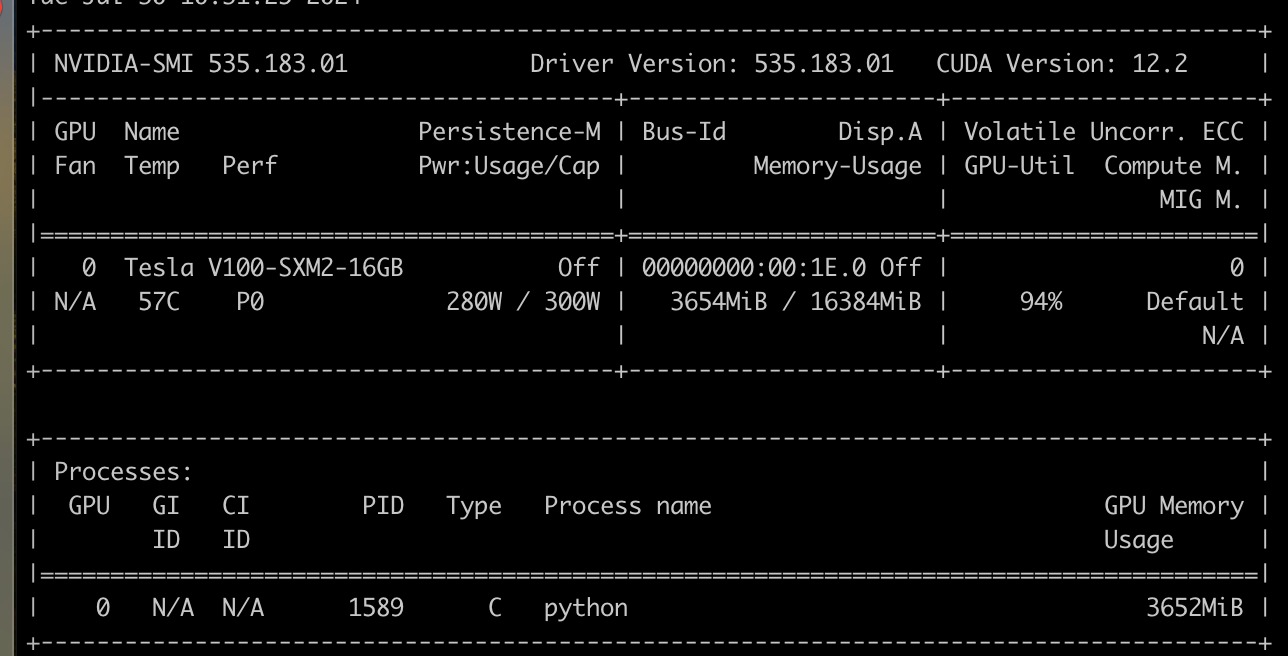


Figure 3: AWS Instances

**Libraries:** Installed necessary libraries like NLTK, Datasets, NumPy, Evaluate, Transformers etc.

## 3.3 Model Selection & Fine-Tuning

We finalised the following two tasks to finetune the models for closed source question answering

* CausalLM
* Sequence-to-Sequence

The models we decided to move ahead with were

**Bart**

* Bart uses a standard seq2seq/machine translation architecture with a bidirectional encoder (like BERT) and a left-to-right decoder (like GPT).
* BART is particularly effective when fine-tuned for text generation.
* We finetuned Bart on CausalLM task.

**T5**

* T5 is an encoder-decoder model pre-trained on a multi-task mixture of unsupervised and supervised tasks and for which each task is converted into a text-to-text format.
* We finetuned Bart on seq2seq task.

**GPT**

* GPT-2 is a transformers model pretrained on a very large corpus of English data in a self-supervised fashion. This means it was pretrained on the raw texts only, with no humans labelling them in any way (which is why it can use lots of publicly available data) with an automatic process to generate inputs and labels from those texts. More precisely, it was trained to guess the next word in sentences.
* We finetuned Bart on CausalLM task.

## 3.4 Evaluation and Testing

The following metrics were used for evaluation of the fine-tuned models.

* ROUGE
* BLEU
* F1-score

# Conclusion