NLP Mini Project Report

NLP Powered Q/A Web App

Submitted in partial fulfilment of the requirements
of the B. E Final Year Semester VIII in degree of
Bachelor of Computer Engineering

BY

Prateek Sharma

Irfan Shaikh

Saurav Kumar

Guide

Prof. Neelam Phadnis

Department Of Computer Engineering

Shree L. R. Tiwari College Of Engineering

Kanakia Park, Mira Road (E), Thane -401 107, Maharashtra.

October 2021

Table of Contents

Introduction

Natural language processing is a huge promising domain. We are utilizing this domain for our text processing project which will be explained in the following lines. The project entails TensorFlow JS Bert model to read an input passage and answer the question asked by user regarding the passage if it is found. Our machine learning algorithm is mounted on a React application and hosted on Azure cloud domain. This project can be used in multiple areas, but it can revolutionize the education sector. It can save time to process a textual paragraph and provide key outputs.

Literature Review

Natural language processing (NLP) is a subfield of linguistics, computer science, and artificial intelligence concerned with the interactions between computers and human language, in particular how to program computers to process and analyze large amounts of natural language data. The goal is a computer capable of "understanding" the contents of documents, including the contextual nuances of the language within them. The technology can then accurately extract information and insights contained in the documents as well as categorize and organize the documents themselves. [1]

Question Answering (QA) is a branch of the Natural Language Understanding (NLU) field (which falls under the NLP umbrella). It aims to implement systems that, given a question in natural language, can extract relevant information from provided data and present it in the form of natural language answer. For example, after being asked, “how warm is it going to be today?” your Siri can extract raw information about today’s temperature from a weather service. In addition, instead of showing it to you as is, it processes the data and presents it to in proper English (or in any other supported language). [2]

BERT is a very powerful natural language processing model that uses transformers for a wide variety of tasks. It works very well for Question Answering, Named Entity Recognition and Natural Language Inference. It was originally presented by Jacob and team from Google in 2018. [3]

Implementation and Results

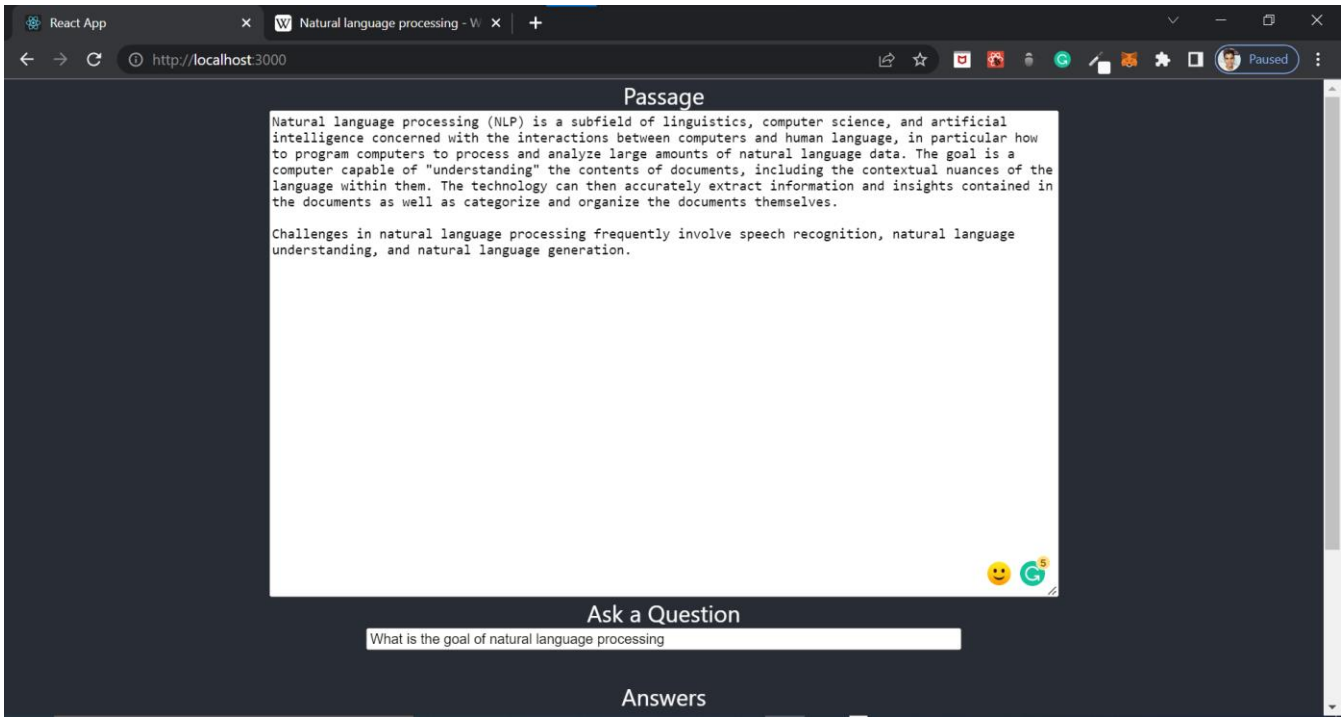
The project was built using react.js and tensorlflow.js predefined bert model. A React.js user interface is provided to capture the passage and query for evalution and scoring. Once the user provides the passage and question real time bert evalution happens and answers are displayed back to the react web app.

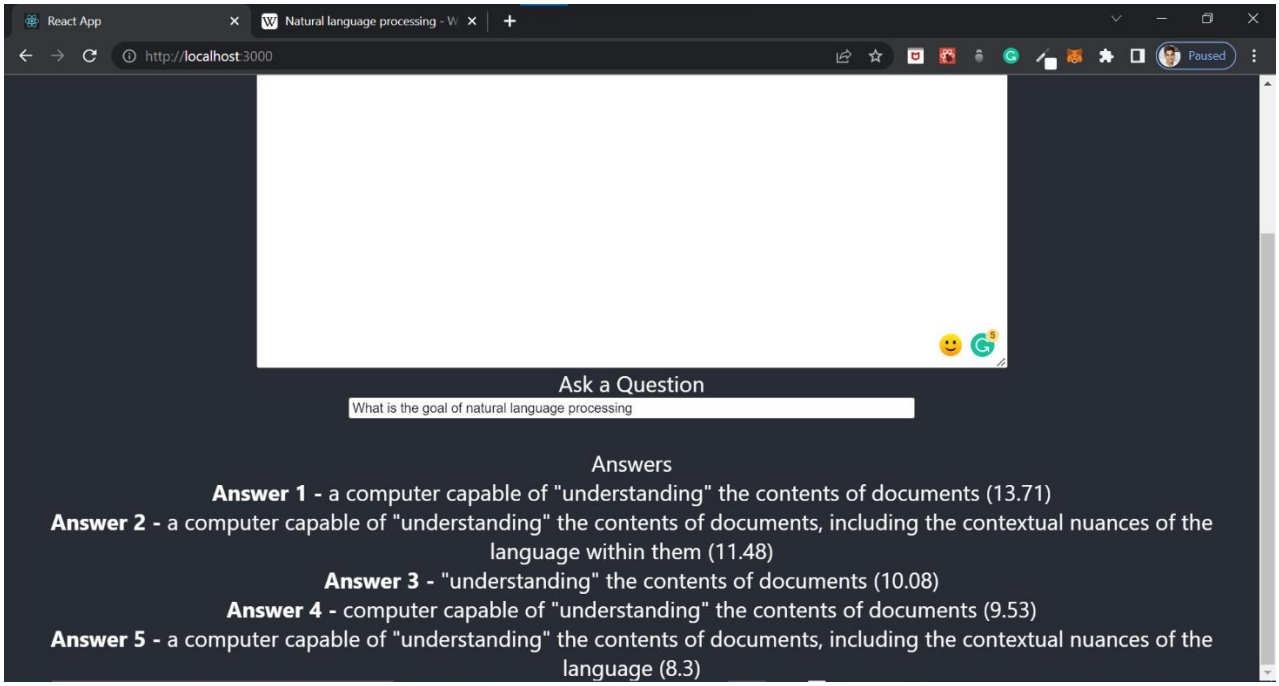
Source Code:

```
import React, {useRef, useEffect, useState} from 'react'
import './App.css'

// 0. Install dependencies
// /npm i @tensorflow/tfjs @tensorflow-models/qna react-loader-spinner

// 1. Import dependencies
import * as tf from "@tensorflow/tfjs"
import * as qna from "@tensorflow-models/qna"
import "react-loader-spinner/dist/loader/css/react-spinner-loader.css"
import Loader from "react-loader-spinner"


const App = () => {

  // 3. Setup references and state hooks
  const passageRef = useRef(null)
  const questionRef = useRef(null)
  const [answer, setAnswer] = useState()
  const [model, setModel] = useState(null)

  // 4. Load Tensorflow Model
  const loadModel = async ()=>{
    const loadedModel = await qna.load()
    setModel(loadedModel)
    console.log('Model loaded.')
  }

  // 5. Handle Questions
  const answerQuestion = async (e) =>{
    if (e.which === 13 && model !== null ){
      console.log('Question submitted.')
      const passage = passageRef.current.value
      const question = questionRef.current.value

      const answers = await model.findAnswers(question, passage)
      setAnswer(answers)
      console.log(answers)
```

```
    }
  }

  useEffect(()=>{loadModel()}, [])

  // 2. Setup input, question and result area
  return (
    <div className="App">
      <header className="App-header">
        {model ==null ?
          <div>
            <div>Model Loading</div>
            <Loader
            type="Puff"
            color="#00BFFF"
            height={100}
            width={100}/>
          </div>
          :
          <React.Fragment>
            Passage
            <textarea ref={passageRef} rows="30" cols="100"></textarea>
            Ask a Question
            <input ref={questionRef} onKeyPress={answerQuestion} size="80"></input>
            <br />
            Answers
            {answer ? answer.map((ans, idx) =><div><b>Answer {idx+1} - </b> {ans.text}
({Math.floor(ans.score*100)/100})</div>) : ""}
            </React.Fragment>
        }
      </header>
    </div>
  );
}

export default App
```

Results:


React App
Natural language processing - W
http://localhost:3000
Paused
Passage
Natural language processing (NLP) is a subfield of linguistics, computer science, and artificial intelligence concerned with the interactions between computers and human language, in particular how to program computers to process and analyze large amounts of natural language data. The goal is a computer capable of "understanding" the contents of documents, including the contextual nuances of the language within them. The technology can then accurately extract information and insights contained in the documents as well as categorize and organize the documents themselves.
Challenges in natural language processing frequently involve speech recognition, natural language understanding, and natural language generation.
Ask a Question
What is the goal of natural language processing
Answers



Conclusion

In this paper, an NLP Powered Q/A Web App is presented built using React.js and Tensorflow.js predefined bert model. When the application first starts the tensorflow.js bert model is loaded. A front end interface is provided to capture the passage and question for processing, Once the passage and query is captured then real time bert evalution and scoring is done and results are displayed back to the react app. Results after the implementation showed that our project worked successfully.

References

[1] “Natural Language Processing – Wiki”. en.wikipedia.org/wiki/Natural_language_processing

[2] “Question Answering (QA) System in Python – Introduction to NLP & a Practical Code Example”. asperbrothers.com/blog/question-answering-python

[3] “Exploring Helpful uses of BERT in your browser with tensorflow.js”. blog.tensorflow.org/2020/03/exploring-helpful-uses-for-bert-in-your-browser-tensorflow-js.html