standford_qa

October 21, 2024

[1]:
```
import pandas as pd
import spacy
# python -m spacy download en_core_web_sm
nlp = spacy.load("en_core_web_sm")
from nltk.stem import SnowballStemmer
stemmer = SnowballStemmer("english")
import matplotlib.pyplot as plt
from pandas import json_normalize
from sklearn.model_selection import train_test_split
from transformers import T5Tokenizer, T5ForConditionalGeneration, Trainer,␣
  ↪TrainingArguments
from datasets import Dataset
```

0.1 Data Collection and Preprocessing:

[3]:
```
# read json data tp json_data
json_data = pd.read_json('data/dev-v1.1.json')

# Use json_normalize to flatten question and id, while keeping answers
df = json_normalize(
    json_data['data'],
    record_path=['paragraphs', 'qas'],
    meta=['title', ['paragraphs', 'context']],
    errors='ignore'
)

# Extract answers and create separate columns for answer1, answer2, answer3
df[['answer1', 'answer2', 'answer3']] = pd.DataFrame(
    df['answers'].apply(lambda ans: [answer['text'] for answer in ans[:3]]).
  ↪to_list(), index=df.index
)

# Drop the original 'answers' column
df = df.drop(columns=['answers'])

# Display the result
df.head()
```

[3]:

```
                                            question  \
0  Which NFL team represented the AFC at Super Bo…
1  Which NFL team represented the NFC at Super Bo…
2                Where did Super Bowl 50 take place?
3                  Which NFL team won Super Bowl 50?
4  What color was used to emphasize the 50th anni…

                         id          title  \
0  56be4db0acb8001400a502ec  Super_Bowl_50
1  56be4db0acb8001400a502ed  Super_Bowl_50
2  56be4db0acb8001400a502ee  Super_Bowl_50
3  56be4db0acb8001400a502ef  Super_Bowl_50
4  56be4db0acb8001400a502f0  Super_Bowl_50

                                  paragraphs.context                  answer1  \
0  Super Bowl 50 was an American football game to…           Denver Broncos
1  Super Bowl 50 was an American football game to…        Carolina Panthers
2  Super Bowl 50 was an American football game to…  Santa Clara, California
3  Super Bowl 50 was an American football game to…           Denver Broncos
4  Super Bowl 50 was an American football game to…                     gold

             answer2                                            answer3
0     Denver Broncos                                     Denver Broncos
1  Carolina Panthers                                  Carolina Panthers
2     Levi's Stadium  Levi's Stadium in the San Francisco Bay Area a…
3     Denver Broncos                                     Denver Broncos
4               gold                                               gold
```

[4]:

```python
# Column that combines questions and context
df['input_text'] = df.apply(lambda row: f"Context: {row['paragraphs.context']}␣
 ↪Question: {row['question']} Answer:", axis=1)

# Split the data into training and validation sets
train_data, val_data = train_test_split(df[['input_text', 'answer1']],␣
 ↪test_size=0.2)

# Convert to list of dictionaries for training
train_data = [{'input_text': row['input_text'], 'target_text': row['answer1']}␣
 ↪for idx, row in train_data.iterrows()]
val_data = [{'input_text': row['input_text'], 'target_text': row['answer1']}␣
 ↪for idx, row in val_data.iterrows()]
```

0.2 Model Training

[ ]:

```
# Initialize the model and tokenizer
model_name = 't5-small'
tokenizer = T5Tokenizer.from_pretrained(model_name)
model = T5ForConditionalGeneration.from_pretrained(model_name)

# Using a dataset object
train_dataset = Dataset.from_list(train_data)
val_dataset = Dataset.from_list(val_data)

# Tokenizing the input and target
def preprocess_data(examples):
    inputs = examples['input_text']
    targets = examples['target_text']

    model_inputs = tokenizer(inputs, max_length=512, truncation=True,␣
  ↪padding="max_length")
    labels = tokenizer(targets, max_length=64, truncation=True,␣
  ↪padding="max_length").input_ids

    # Replacing padding token ids in labels with -100 to ignore them
    labels_with_padding = [-100 if token == tokenizer.pad_token_id else token␣
  ↪for token in labels]
    model_inputs['labels'] = labels_with_padding
    return model_inputs

# Applying preprocessing
train_dataset = train_dataset.map(preprocess_data, batched=True)
val_dataset = val_dataset.map(preprocess_data, batched=True)

training_args = TrainingArguments(
    output_dir='./results',
    evaluation_strategy="steps",
    per_device_train_batch_size=4,
    per_device_eval_batch_size=4,
    num_train_epochs=1,
    save_steps=10,
    eval_steps=10,
    logging_dir='./logs',
    logging_steps=10,
)

trainer = Trainer(
    model=model,
    args=training_args,
    train_dataset=train_dataset,
    eval_dataset=val_dataset,
)

# Start training
trainer.train(resume_from_checkpoint=True)

# Save the model
trainer.save_model('./trained_model')
tokenizer.save_pretrained('./trained_model')
```

0.3 Model Evaluation

[5]:

```
# Load the trained model and tokenizer
model_path = './app/trained_model'
tokenizer = T5Tokenizer.from_pretrained(model_path)
model = T5ForConditionalGeneration.from_pretrained(model_path)

# Prepare the validation dataset for evaluation (assuming `val_data` is already␣
 ↪prepared)
val_dataset = Dataset.from_list(val_data)

# Tokenizing the input and target
def preprocess_data(examples):
    inputs = examples['input_text']
    targets = examples['target_text']

    model_inputs = tokenizer(inputs, max_length=512, truncation=True,␣
 ↪padding="max_length")
    labels = tokenizer(targets, max_length=64, truncation=True,␣
 ↪padding="max_length").input_ids

    # Replacing padding token ids in labels with -100 to ignore them during␣
 ↪loss computation
    labels_with_padding = [-100 if token == tokenizer.pad_token_id else token␣
 ↪for token in labels]
    model_inputs['labels'] = labels_with_padding
    return model_inputs

# Applying preprocessing on validation dataset
val_dataset = val_dataset.map(preprocess_data, batched=True)

# Define evaluation arguments
eval_args = TrainingArguments(
    output_dir='./results',
    per_device_eval_batch_size=4,
    logging_dir='./logs',
    logging_steps=10,
)

# Create a trainer instance
trainer = Trainer(
    model=model,
    args=eval_args,
    eval_dataset=val_dataset,
)

# Run evaluation
eval_results = trainer.evaluate()

# Print evaluation results
print(f"Evaluation Results: {eval_results}")

# Generate predictions for the validation set
def generate_answer(input_text):
    inputs = tokenizer(input_text, return_tensors="pt", max_length=512,␣
 ↪truncation=True)
    input_ids = inputs.input_ids.to(model.device)

    # Generate output
    outputs = model.generate(input_ids, max_length=64, num_beams=4,␣
 ↪early_stopping=True)

    # Decode the generated output
    answer = tokenizer.decode(outputs[0], skip_special_tokens=True)
    return answer

# Test the model on some validation samples
for i in range(5):  # Adjust to test more examples
    sample = val_data[i]
    input_text = sample['input_text']
    target_text = sample['target_text']

    predicted_answer = generate_answer(input_text)
    print(f"Input: {input_text}")
    print(f"Predicted Answer: {predicted_answer}")
    print(f"Actual Answer: {target_text}")
    print("-" * 50)
```

```
Map:   0%|          | 0/2114 [00:00<?, ? examples/s]

  0%|          | 0/529 [00:00<?, ?it/s]

Evaluation Results: {'eval_loss': 0.029073098674416542,
'eval_model_preparation_time': 0.0016, 'eval_runtime': 70.6523,
'eval_samples_per_second': 29.921, 'eval_steps_per_second': 7.487}
```

```
Input: Context: The Rankine cycle is the fundamental thermodynamic underpinning
of the steam engine. The cycle is an arrangement of components as is typically
used for simple power production, and utilizes the phase change of water
(boiling water producing steam, condensing exhaust steam, producing liquid
water)) to provide a practical heat/power conversion system. The heat is
supplied externally to a closed loop with some of the heat added being converted
to work and the waste heat being removed in a condenser. The Rankine cycle is
used in virtually all steam power production applications. In the 1990s, Rankine
steam cycles generated about 90% of all electric power used throughout the
world, including virtually all solar, biomass, coal and nuclear power plants. It
is named after William John Macquorn Rankine, a Scottish polymath. Question:
What happens to waste heat in the Rankine cycle? Answer:
Predicted Answer: removed in a condenser
Actual Answer: removed in a condenser
--------------------------------------------------
Input: Context: In some rural areas in the United Kingdom, there are dispensing
physicians who are allowed to both prescribe and dispense prescription-only
medicines to their patients from within their practices. The law requires that
the GP practice be located in a designated rural area and that there is also a
specified, minimum distance (currently 1.6 kilometres) between a patient's home
and the nearest retail pharmacy. This law also exists in Austria for general
physicians if the nearest pharmacy is more than 4 kilometers away, or where none
is registered in the city. Question: What is the minimum distance between a
patient's home and the nearest pharmacy that allows a physician in Austria to
give out medicine? Answer:
Predicted Answer: 1.6 kms
Actual Answer: more than 4 kilometers
--------------------------------------------------
Input: Context: The V&A holds over 19,000 items from the Islamic world, ranging
from the early Islamic period (the 7th century) to the early 20th century. The
Jameel Gallery of Islamic Art, opened in 2006, houses a representative display
of 400 objects with the highlight being the Ardabil Carpet, the centrepiece of
the gallery. The displays in this gallery cover objects from Spain, North
Africa, the Middle East, Central Asia and Afghanistan. A masterpiece of Islamic
art is a 10th-century Rock crystal ewer. Many examples of Qur'āns with exquisite
calligraphy dating from various periods are on display. A 15th-century minbar
from a Cairo mosque with ivory forming complex geometrical patterns inlaid in
wood is one of the larger objects on display. Extensive examples of ceramics
especially Iznik pottery, glasswork including 14th-century lamps from mosques
and metalwork are on display. The collection of Middle Eastern and Persian rugs
and carpets is amongst the finest in the world, many were part of the Salting
Bequest of 1909. Examples of tile work from various buildings including a
fireplace dated 1731 from Istanbul made of intricately decorated blue and white
tiles and turquoise tiles from the exterior of buildings from Samarkand are also
displayed. Question: What is considered the centerpiece of the Jameel Gallery of
Islamic Art? Answer:
Predicted Answer: Ardabil Carpet
Actual Answer: Ardabil Carpet
```

```
--------------------------------------------------
Input: Context: Various gold-themed promotions and initiatives were held
throughout the 2015 NFL season to tie into the "Golden Super Bowl"; gold-tinted
logos were implemented across the NFL's properties and painted on fields, the
numbering of the 50-yard line on fields was colored gold, and beginning on week
7, all sideline jackets and hats featured gold-trimmed logos. Gold footballs
were given to each high school that has had a player or coach appear in the
Super Bowl, and "homecoming" events were also held by Super Bowl-winning teams
at games. Question: What was given to high schools where former students went on
to play or coach in a Super Bowl? Answer:
Predicted Answer: Gold footballs
Actual Answer: Gold footballs
--------------------------------------------------
Input: Context: What we now call gravity was not identified as a universal force
until the work of Isaac Newton. Before Newton, the tendency for objects to fall
towards the Earth was not understood to be related to the motions of celestial
objects. Galileo was instrumental in describing the characteristics of falling
objects by determining that the acceleration of every object in free-fall was
constant and independent of the mass of the object. Today, this acceleration due
to gravity towards the surface of the Earth is usually designated as  and has a
magnitude of about 9.81 meters per second squared (this measurement is taken
from sea level and may vary depending on location), and points toward the center
of the Earth. This observation means that the force of gravity on an object at
the Earth's surface is directly proportional to the object's mass. Thus an
object that has a mass of  will experience a force: Question: Where was the
measurment for the standard gravity on Earth taken? Answer:
Predicted Answer: sea level
Actual Answer: sea level
--------------------------------------------------
```

0.4 Conclusion

0.4.1 Methodology

The task involved fine-tuning a T5-based model for question answering using a dataset containing contexts, questions, and answers. The steps for training and evaluation were as follows:

Data Preparation:

- A JSON dataset was processed and flattened to extract contexts, questions, and answers.
- The data was further split into training and validation sets, where each row contained an input_text (combining context and question) and a target (answer1).

Model and Tokenization:

- The T5Tokenizer and T5ForConditionalGeneration model were used from the Hugging Face transformers library. Both the training and validation datasets were tokenized, with inputs truncated to a maximum length of 512 tokens and target sequences to 64 tokens.

Model Setup and Training:

- The Trainer class was employed for training, using a batch size of 4, with evaluation strategy set to run every 10 steps. Training was resumed from a checkpoint. The model was trained for 1 epoch with num_train_epochs=1.

0.4.2 Evaluation

- We evaluated the model was evaluated using the validation dataset after training, calculating the loss and checking predictions against actual answers.

0.4.3 Results

- Evaluation Loss: The model achieved a very low evaluation loss of 0.029.
- Efficiency: The model processed 29.92 samples per second and 7.49 steps per second during evaluation.
- Example Prediction: On one sample from the validation set, the model was given the context of the Rankine cycle and asked, “What happens to waste heat in the Rankine cycle?”
  - Predicted Answer: “removed in a condenser”
  - Actual Answer: “removed in a condenser”

The model’s performance on this example was highly accurate, demonstrating effective learning and prediction capabilities for T5 model for the question-answering task.