Module 5: Assignment

Submitted By:

Group 5

Dennis Darko, Harsh Dodiya, Romin Patel, Hetvi Shah

Master of Professional Studies in Analytics, Northeastern University, Vancouver

EAI 6010, 90618: Applications of Artificial Intelligence

Prof. Richard Munthali

14st August, 2024

Abstract

This report presents the development and deployment of a microservice that classifies news articles into predefined categories using a machine learning model trained on the AG News dataset. The microservice, deployed on Google Cloud Run, leverages containerization to handle dependencies and resource constraints efficiently. The AWD_LSTM ULMFiT model achieved an overall accuracy of 88% during validation, with strong precision, recall, and F1-scores across all categories. This document outlines the dataset selection, model parameters, service deployment, challenges encountered, and examples of the service's input and output.

Keywords: AG News, text classification, microservice, Google Cloud Run, deployment, machine learning

Main Section

1. Introduction

In today's digital landscape, the ability to automatically categorize news articles is crucial for enhancing user experience on news platforms, enabling content filtering, and conducting sentiment analysis. Machine learning models have significantly improved the efficiency and accuracy of text classification tasks. This report details the deployment of a microservice that classifies news articles into four categories—World, Sports, Business, and Sci/Tech—using a model trained on the AG News dataset. The service was developed in Python, containerized using Docker, and deployed on Google Cloud Run, ensuring scalability and accessibility.

2. Business Problems

The need for automating the classification of news articles has become increasingly important for news platforms and aggregators. Manual categorization is time-consuming and prone to errors, leading to inconsistent user experiences. The AG News Category Prediction Microservice aims to solve the following business problems:

- **News Categorization:** The microservice automates the categorization of news articles into predefined categories, which is essential for organizing content on news websites and aggregators.
- **Content Filtering:** The service can be used to filter and recommend news articles based on user preferences, enhancing personalized content delivery.
- **Sentiment Analysis:** Although not the primary focus of this service, the model could be adapted for sentiment analysis within different news categories, providing insights into public opinion and media trends.

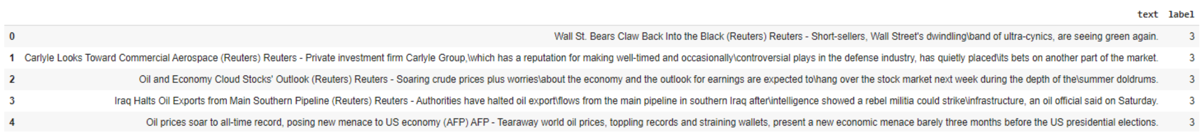
3. Dataset Selection and Parameters

3.1 Dataset Selection

The AG News dataset was selected for this project due to its structured format and widespread use in text classification tasks. The dataset comprises news articles categorized into four classes:

- World
- Sports
- Business
- Sci/Tech.

This balanced distribution of categories provides a robust foundation for training models to achieve high accuracy in predicting the category of news articles.



	text	label
0	Wall St. Bears Claw Back Into the Black (Reuters) Reuters - Short-sellers, Wall Street's dwindling\band of ultra-cynics, are seeing green again.	3
1	Carlyle Looks Toward Commercial Aerospace (Reuters) Reuters - Private investment firm Carlyle Group,\which has a reputation for making well-timed and occasionally\controversial plays in the defense industry, has quietly placed\its bets on another part of the market.	3
2	Oil and Economy Cloud Stocks' Outlook (Reuters) Reuters - Soaring crude prices plus worries\about the economy and the outlook for earnings are expected to\hang over the stock market next week during the depth of the\summer doldrums.	3
3	Iraq Halts Oil Exports from Main Southern Pipeline (Reuters) Reuters - Authorities have halted oil export\flows from the main pipeline in southern Iraq after\intelligence showed a rebel militia could strike\infrastructure, an oil official said on Saturday.	3
4	Oil prices soar to all-time record, posing new menace to US economy (AFP) AFP - Tearaway world oil prices, toppling records and straining wallets, present a new economic menace barely three months before the US presidential elections.	3

Figure 1: Dataset

3.2 Constraints on the Dataset

To manage computational resources effectively and expedite the training process, the dataset was constrained to a 10% subset of the original training and validation data. The dataset contains 96,000 observations, and 10% of this dataset (9,600 observations) was used. This subset was sufficient to evaluate the model's performance while significantly reducing training time and resource consumption.

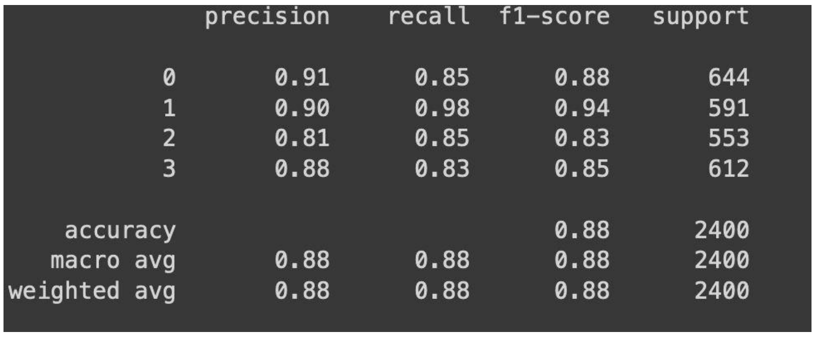
3.3 Parameter Selection

The model's performance was optimized by fine-tuning key parameters during the training process:

- **Learning Rate:** The learning rate was carefully adjusted to ensure stable convergence during training, avoiding the risks of overshooting or slow convergence.
- **Batch Size:** A moderate batch size was chosen to balance computational load and memory usage while maintaining sufficient gradient updates per iteration.
- **Number of Epochs:** The number of training epochs was determined through experimentation, aiming to achieve the best trade-off between training time and model performance.

4. Model Validation Results

Here are the classification report and confusion matrix:



```
              precision    recall  f1-score   support

           0       0.91      0.85      0.88       644
           1       0.90      0.98      0.94       591
           2       0.81      0.85      0.83       553
           3       0.88      0.83      0.85       612

    accuracy                           0.88      2400
   macro avg       0.88      0.88      0.88      2400
weighted avg       0.88      0.88      0.88      2400
```

Figure 2: Classification Report

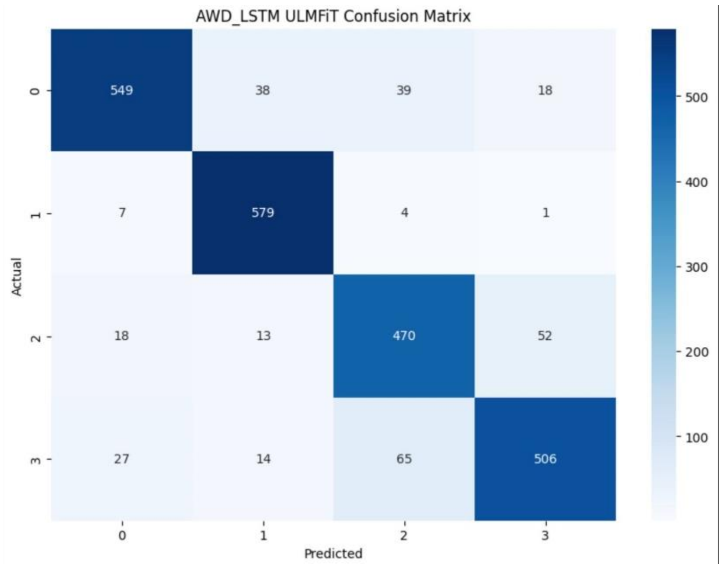


Figure 3: Confusion Matrix

5. Service Description

The deployed microservice provides an API that allows users to input the text of a news article and receive a prediction of its category. The service is designed to handle text classification tasks efficiently, making it suitable for news aggregation platforms and content management systems.

5.1 General Input and Output

- Input: Plain text of a news article.
- Output: Predicted category label (World, Sports, Business, Sci/Tech).

The service processes the input text, applies the trained model, and returns the most likely category.

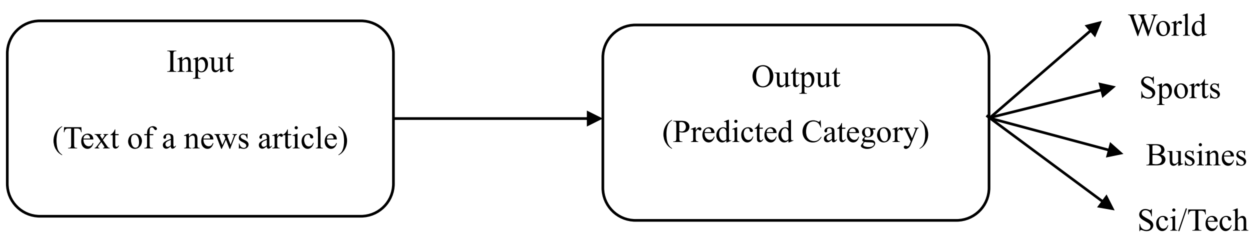


Figure 4: Service Process

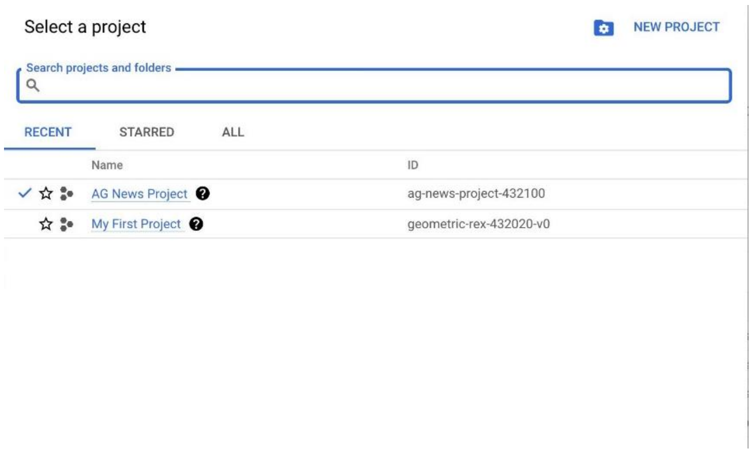
6. Service Deployment

6.1 Deployment Platform: Google Cloud Run

The service was deployed using Google Cloud Run, a fully managed compute platform that automatically scales applications based on incoming requests. Cloud Run was chosen for its ability to handle containerized applications, making it ideal for deploying machine learning models with complex dependencies.

6.2 Project Setup

Google Cloud Project Creation: A new Google Cloud project with the ID ag-news-project-432100 was created to manage the deployment and associated resources.

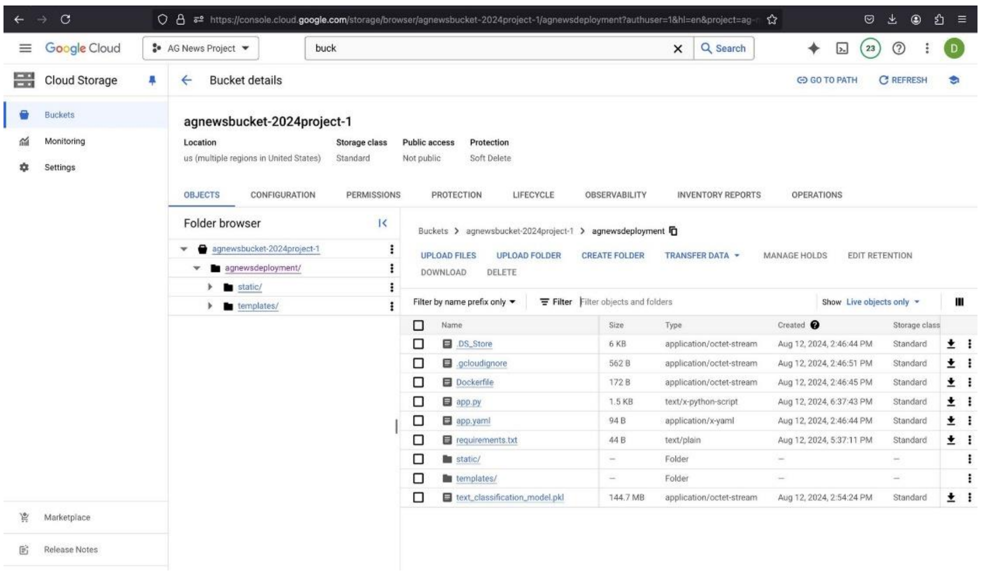


Name	ID
AG News Project	ag-news-project-432100
My First Project	geometric-rex-432020-v0

Figure 5: Google Cloud Project Creation

6.3 Cloud Storage

A Cloud Storage bucket named agnewsbucket-2024project-1 was created to store the trained model and related files. This bucket serves as a repository for deployment assets, ensuring easy access during the deployment process.



https://console.cloud.google.com/storage/browser/agnewsbucket-2024project-1/agnewsdeployment?authuser=1&hl=en&project=ag-

agnewsbucket-2024project-1

Location	Storage class	Public access	Protection
us (multiple regions in United States)	Standard	Not public	Soft Delete

agnewsbucket-2024project-1
agnewsdeployment/
static/
templates/

Buckets > agnewsbucket-2024project-1 > agnewsdeployment

Name	Size	Type	Created	Storage class
.DS_Store	6 KB	application/octet-stream	Aug 12, 2024, 2:46:44 PM	Standard
.gcloudignore	562 B	application/octet-stream	Aug 12, 2024, 2:46:51 PM	Standard
Dockerfile	172 B	application/octet-stream	Aug 12, 2024, 2:46:45 PM	Standard
app.py	1.5 KB	text/x-python-script	Aug 12, 2024, 6:37:43 PM	Standard
app.yaml	94 B	application/x-yaml	Aug 12, 2024, 2:46:44 PM	Standard
requirements.txt	44 B	text/plain	Aug 12, 2024, 5:37:11 PM	Standard
static/	–	Folder	–	–
templates/	–	Folder	–	–
text_classification_model.pkl	144.7 MB	application/octet-stream	Aug 12, 2024, 2:54:24 PM	Standard

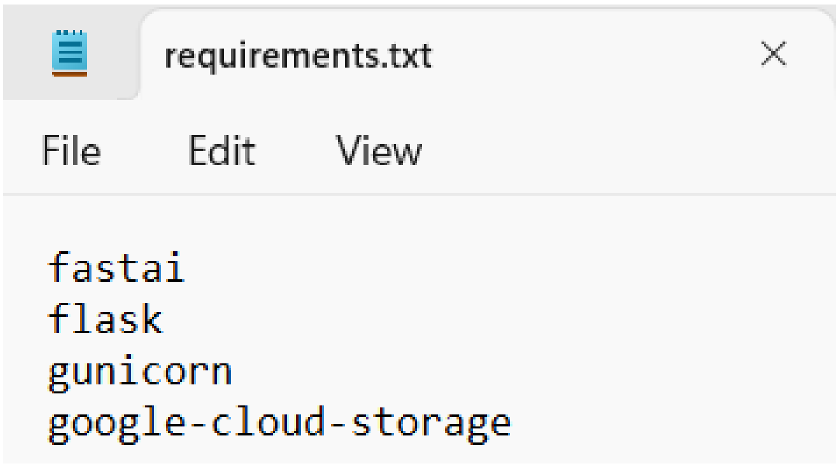
Figure 6: Bucket Creation

6.4 Command Line Tool Setup

The Google Cloud SDK was installed to facilitate the deployment process. This setup was necessary because Cloud Run does not provide a convenient user interface for building and deploying services. The SDK was configured on a macOS system, requiring adjustments to the system's PATH environment variable to include Homebrew, the package manager used for installing the SDK.

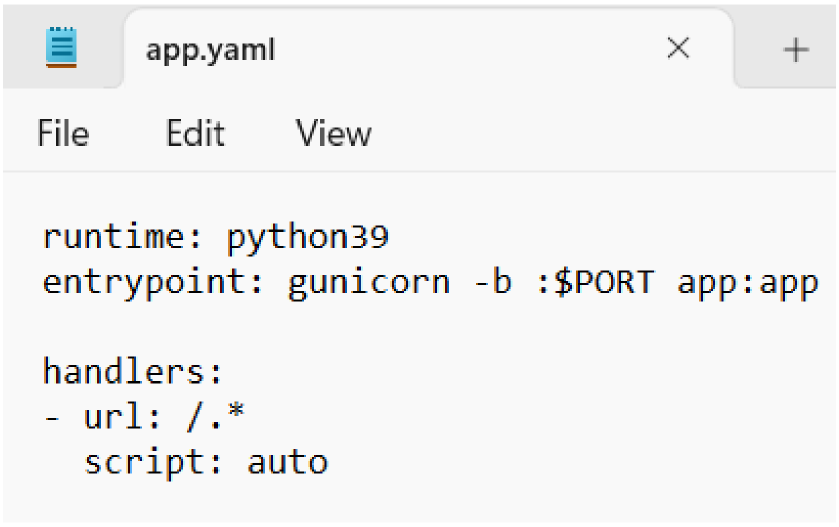
6.5 Service Code Development

The service was developed using Python, with the main application logic implemented in app.py. The service's dependencies, including the machine learning libraries, were listed in requirements.txt. A Dockerfile was created to define the environment in which the service would run, specifying the base image, dependencies, and commands for running the service.



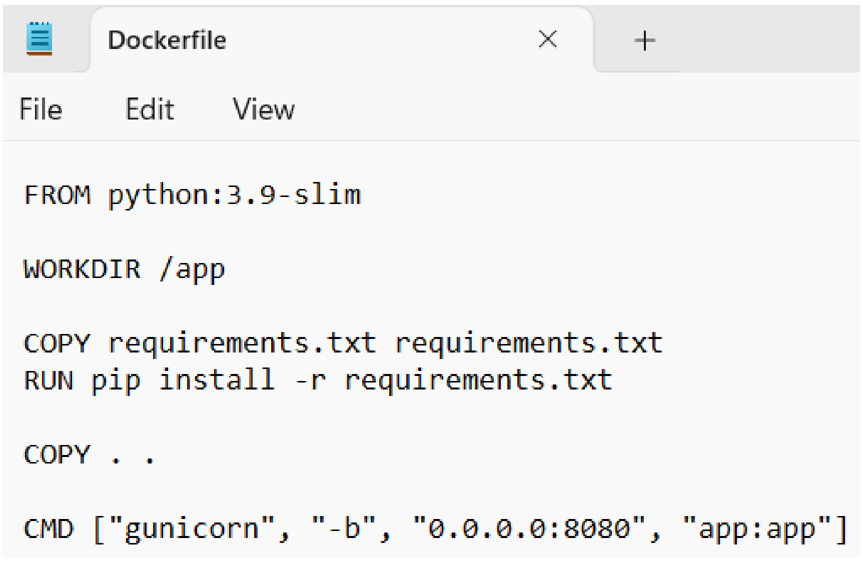
```
fastai
flask
gunicorn
google-cloud-storage
```

Figure 7: Requirement File



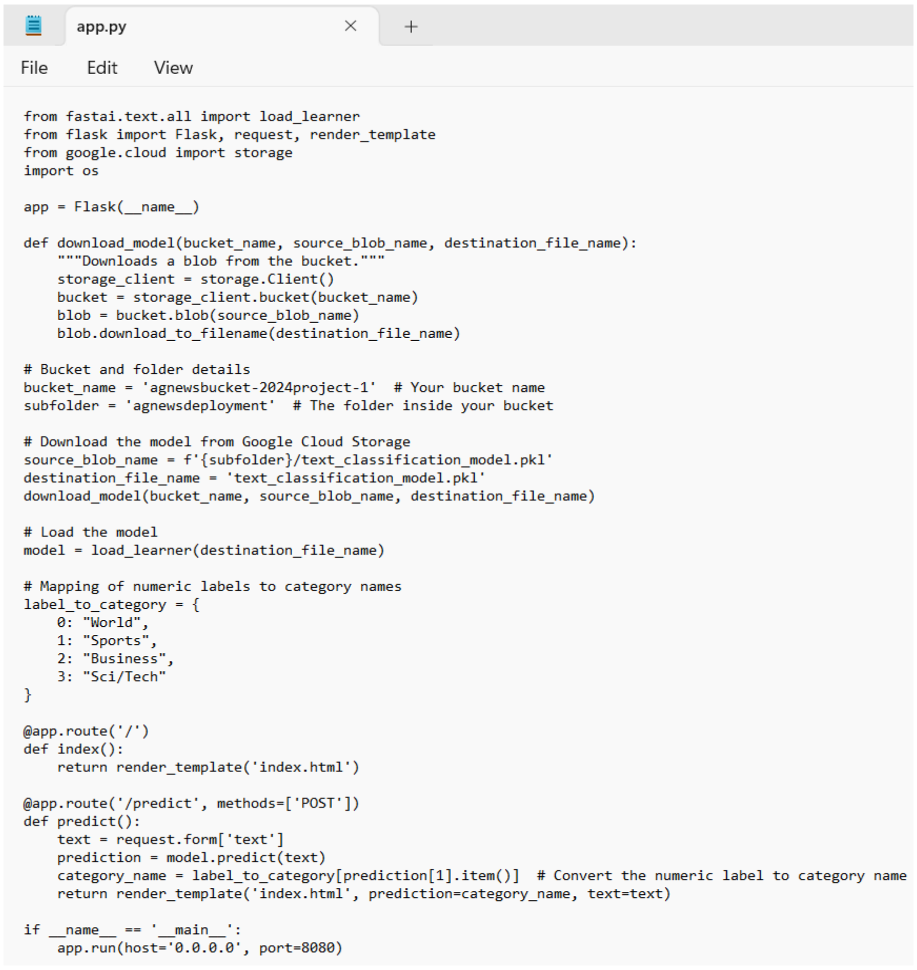
```
runtime: python39
entrypoint: gunicorn -b :$PORT app:app

handlers:
- url: /.*
  script: auto
```

Figure 8: YAML File



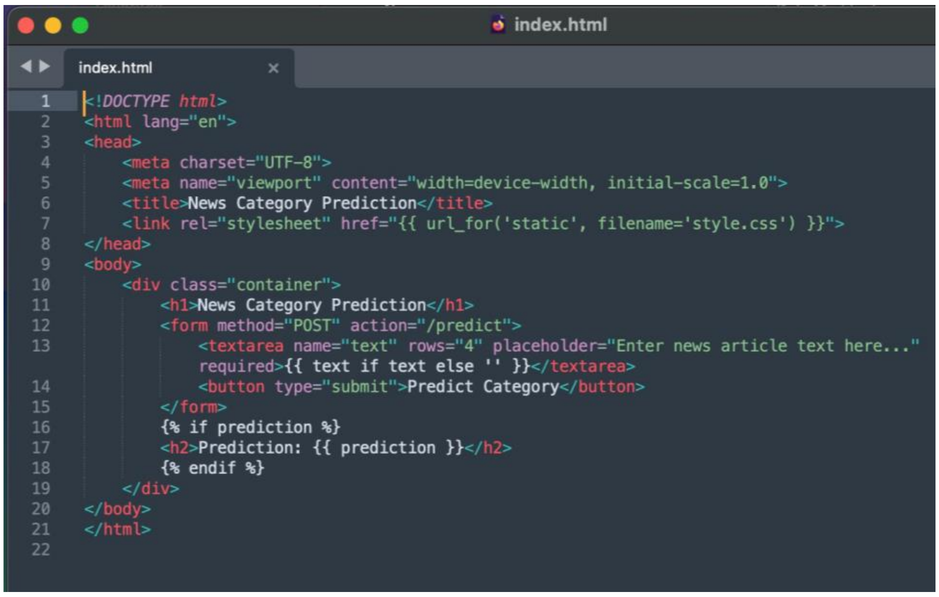
```
FROM python:3.9-slim

WORKDIR /app

COPY requirements.txt requirements.txt
RUN pip install -r requirements.txt

COPY . .

CMD ["gunicorn", "-b", "0.0.0.0:8080", "app:app"]
```

Figure 9: Docker File



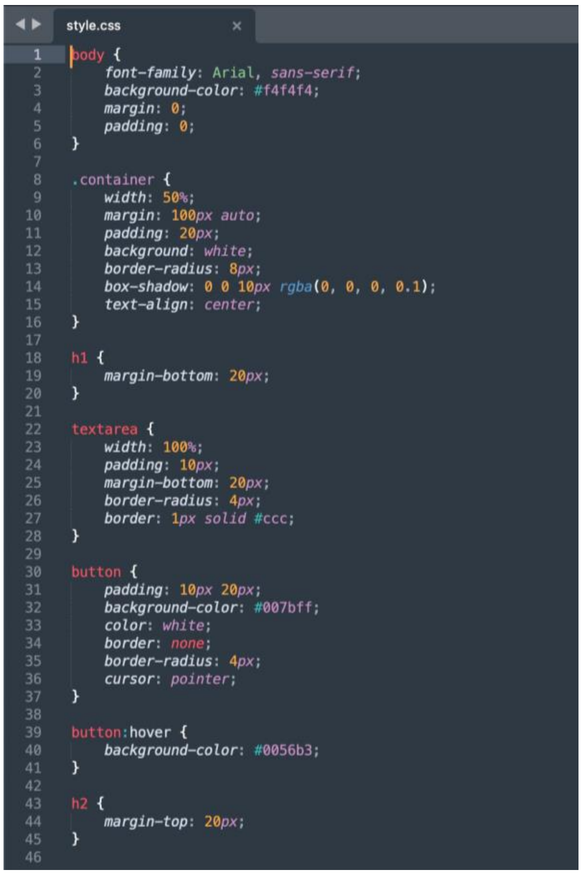
```python
from fastai.text.all import load_learner
from flask import Flask, request, render_template
from google.cloud import storage
import os

app = Flask(__name__)

def download_model(bucket_name, source_blob_name, destination_file_name):
    """Downloads a blob from the bucket."""
    storage_client = storage.Client()
    bucket = storage_client.bucket(bucket_name)
    blob = bucket.blob(source_blob_name)
    blob.download_to_filename(destination_file_name)

# Bucket and folder details
bucket_name = 'agnewsbucket-2024project-1'  # Your bucket name
subfolder = 'agnewsdeployment'  # The folder inside your bucket

# Download the model from Google Cloud Storage
source_blob_name = f'{subfolder}/text_classification_model.pkl'
destination_file_name = 'text_classification_model.pkl'
download_model(bucket_name, source_blob_name, destination_file_name)

# Load the model
model = load_learner(destination_file_name)

# Mapping of numeric labels to category names
label_to_category = {
    0: "World",
    1: "Sports",
    2: "Business",
    3: "Sci/Tech"
}

@app.route('/')
def index():
    return render_template('index.html')

@app.route('/predict', methods=['POST'])
def predict():
    text = request.form['text']
    prediction = model.predict(text)
    category_name = label_to_category[prediction[1].item()]  # Convert the numeric label to category name
    return render_template('index.html', prediction=category_name, text=text)

if __name__ == '__main__':
    app.run(host='0.0.0.0', port=8080)
```

Figure 10: Application File



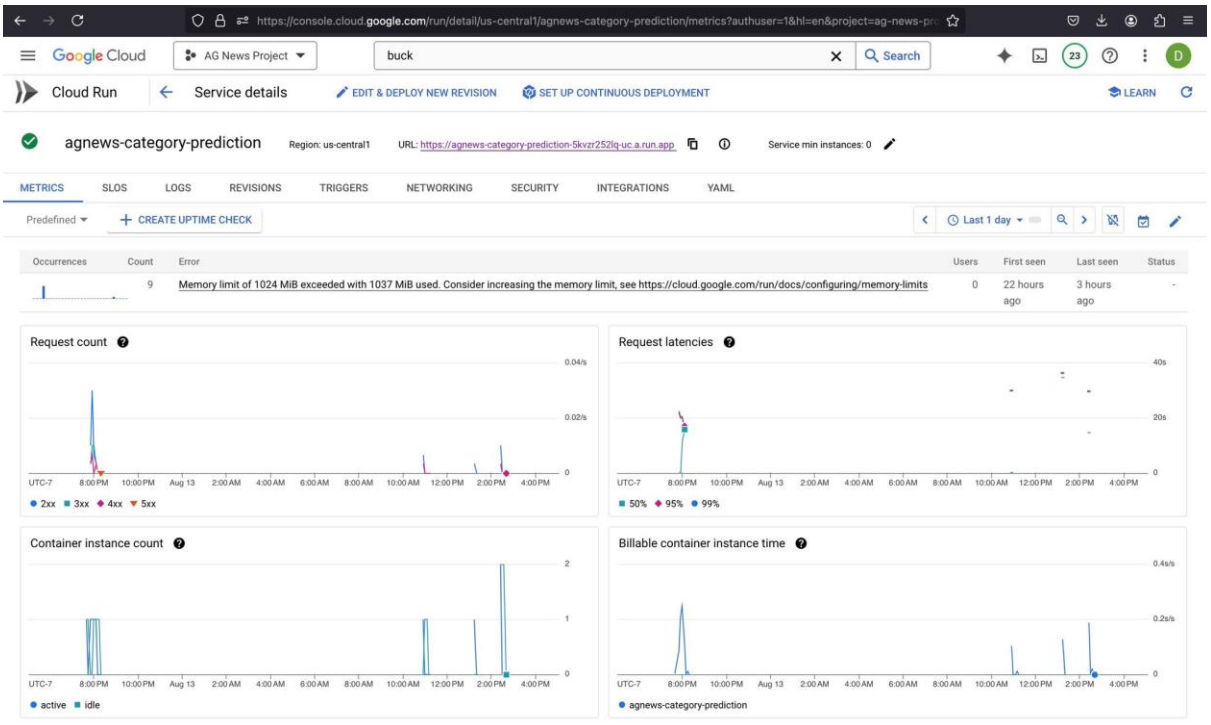
```html
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>News Category Prediction</title>
    <link rel="stylesheet" href="{{ url_for('static', filename='style.css') }}">
</head>
<body>
    <div class="container">
        <h1>News Category Prediction</h1>
        <form method="POST" action="/predict">
            <textarea name="text" rows="4" placeholder="Enter news article text here..."
            required>{{ text if text else '' }}</textarea>
            <button type="submit">Predict Category</button>
        </form>
        {% if prediction %}
        <h2>Prediction: {{ prediction }}</h2>
        {% endif %}
    </div>
</body>
</html>
```

Figure 11: HTML File



```css
body {
    font-family: Arial, sans-serif;
    background-color: #f4f4f4;
    margin: 0;
    padding: 0;
}

.container {
    width: 50%;
    margin: 100px auto;
    padding: 20px;
    background: white;
    border-radius: 8px;
    box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);
    text-align: center;
}

h1 {
    margin-bottom: 20px;
}

textarea {
    width: 100%;
    padding: 10px;
    margin-bottom: 20px;
    border-radius: 4px;
    border: 1px solid #ccc;
}

button {
    padding: 10px 20px;
    background-color: #007bff;
    color: white;
    border: none;
    border-radius: 4px;
    cursor: pointer;
}

button:hover {
    background-color: #0056b3;
}

h2 {
    margin-top: 20px;
}

```

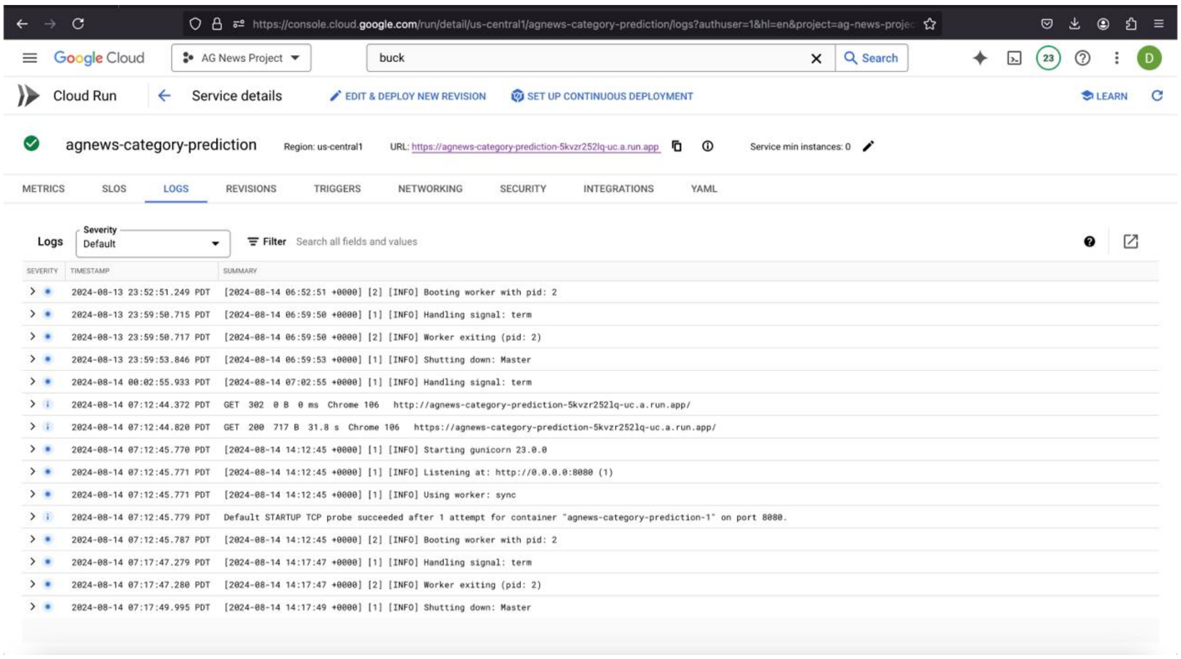
Figure 12: CSS File



agnews-category-prediction Region: us-central1 URL: https://agnews-category-prediction-5kvzr252lq-uc.a.run.app Service min instances: 0

Occurrences	Count	Error	Users	First seen	Last seen	Status
	9	Memory limit of 1024 MiB exceeded with 1037 MiB used. Consider increasing the memory limit, see https://cloud.google.com/run/docs/configuring/memory-limits	0	22 hours ago	3 hours ago	-

Figure 13: Bucket


https://console.cloud.google.com/run/detail/us-central1/agnews-category-prediction/logs?authuser=1&hl=en&project=ag-news-projec
Google Cloud
AG News Project
buck
Search
Cloud Run
Service details
EDIT & DEPLOY NEW REVISION
SET UP CONTINUOUS DEPLOYMENT
LEARN
agnews-category-prediction
Region: us-central1
URL: https://agnews-category-prediction-5kvzr252lq-uc.a.run.app
Service min instances: 0
METRICS
SLOS
LOGS
REVISIONS
TRIGGERS
NETWORKING
SECURITY
INTEGRATIONS
YAML
Logs
Severity
Default
Filter
Search all fields and values

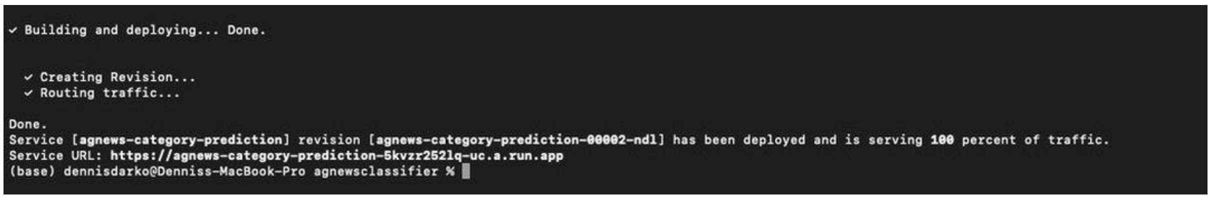
```
SEVERITY  TIMESTAMP                    SUMMARY
>         2024-08-13 23:52:51.249 PDT  [2024-08-14 06:52:51 +0000] [2] [INFO] Booting worker with pid: 2
>         2024-08-13 23:59:50.715 PDT  [2024-08-14 06:59:50 +0000] [1] [INFO] Handling signal: term
>         2024-08-13 23:59:50.717 PDT  [2024-08-14 06:59:50 +0000] [2] [INFO] Worker exiting (pid: 2)
>         2024-08-13 23:59:53.846 PDT  [2024-08-14 06:59:53 +0000] [1] [INFO] Shutting down: Master
>         2024-08-14 00:02:55.933 PDT  [2024-08-14 07:02:55 +0000] [1] [INFO] Handling signal: term
>         2024-08-14 07:12:44.372 PDT  GET  302  0 B  0 ms  Chrome 106   http://agnews-category-prediction-5kvzr252lq-uc.a.run.app/
>         2024-08-14 07:12:44.820 PDT  GET  200  717 B  31.8 s  Chrome 106   https://agnews-category-prediction-5kvzr252lq-uc.a.run.app/
>         2024-08-14 07:12:45.770 PDT  [2024-08-14 14:12:45 +0000] [1] [INFO] Starting gunicorn 23.0.0
>         2024-08-14 07:12:45.771 PDT  [2024-08-14 14:12:45 +0000] [1] [INFO] Listening at: http://0.0.0.0:8080 (1)
>         2024-08-14 07:12:45.771 PDT  [2024-08-14 14:12:45 +0000] [1] [INFO] Using worker: sync
>         2024-08-14 07:12:45.779 PDT  Default STARTUP TCP probe succeeded after 1 attempt for container "agnews-category-prediction-1" on port 8080.
>         2024-08-14 07:12:45.787 PDT  [2024-08-14 14:12:45 +0000] [2] [INFO] Booting worker with pid: 2
>         2024-08-14 07:17:47.279 PDT  [2024-08-14 14:17:47 +0000] [1] [INFO] Handling signal: term
>         2024-08-14 07:17:47.280 PDT  [2024-08-14 14:17:47 +0000] [2] [INFO] Worker exiting (pid: 2)
>         2024-08-14 07:17:49.995 PDT  [2024-08-14 14:17:49 +0000] [1] [INFO] Shutting down: Master
```

Figure 14: logs

6.6 Containerization and Deployment

The service was containerized using Docker, allowing it to be deployed as a scalable microservice on Google Cloud Run. The deployment was executed using the following command:

gcloud run deploy agnews-category-prediction --project ag-news-project-432100 --source . --region us-central1 --allow-unauthenticated --memory=1024Mi

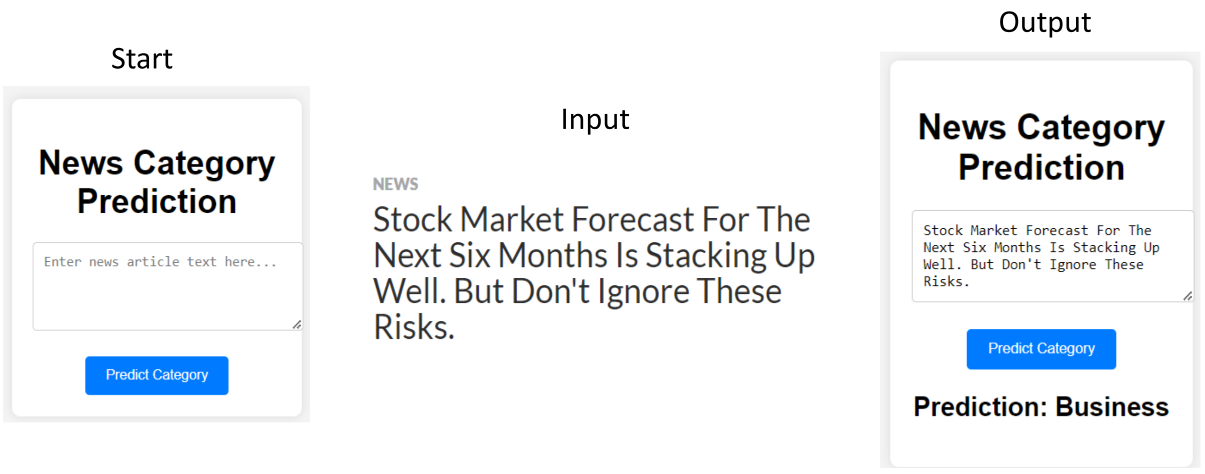


```
✓ Building and deploying... Done.


  ✓ Creating Revision...
  ✓ Routing traffic...

Done.
Service [agnews-category-prediction] revision [agnews-category-prediction-00002-ndl] has been deployed and is serving 100 percent of traffic.
Service URL: https://agnews-category-prediction-5kvzr252lq-uc.a.run.app
(base) dennisdarko@Denniss-MacBook-Pro agnewsclassifier %
```

Figure 15: Deployment

7. Service Examples of Service's Input and Output

- Link: https://agnews-category-prediction-5kvzr252lq-uc.a.run.app/
- The microservice was tested with various news articles. Here are some examples:



Enter news article text here...

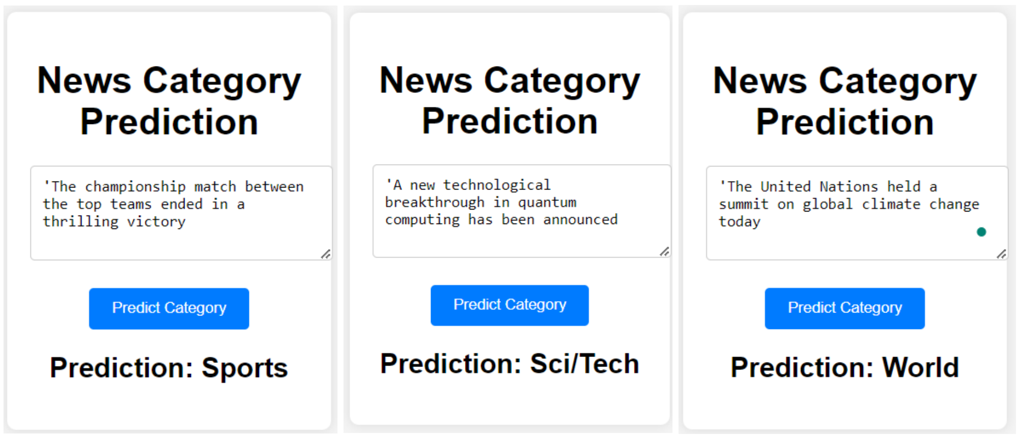
NEWS

Stock Market Forecast For The Next Six Months Is Stacking Up Well. But Don't Ignore These Risks.

Stock Market Forecast For The Next Six Months Is Stacking Up Well. But Don't Ignore These Risks.

Prediction: Business

Figure 16: Prediction Example 1



'The championship match between the top teams ended in a thrilling victory

Prediction: Sports

'A new technological breakthrough in quantum computing has been announced

Prediction: Sci/Tech

'The United Nations held a summit on global climate change today

Prediction: World

Figure 17: Example 2-4

8. Challenges Across Operating Systems

Windows OS:

- Issue: Despite successful deployment of all required files, the web application consistently displayed a "Service Unavailable" error when accessing the URL.
- Impact: This issue hindered the ability to test and verify the deployment on Windows OS, leading to delays in the deployment timeline.

macOS:

- Issue: Significant challenges were encountered during the installation of the SDK, particularly with the Homebrew package manager.
- Details: Manual intervention was required to add Homebrew to the system's PATH environment variable, causing delays in the deployment process.
- Resolution: After resolving the PATH issue, the service was successfully deployed on macOS.

9. CONCLUSION

In this report, we successfully developed and deployed a microservice that automates the categorization of news articles into predefined categories using a machine learning model trained on the AG News dataset. The service, which was containerized and deployed on Google Cloud Run, achieved an accuracy of 88% during validation, demonstrating the effectiveness of the AWD_LSTM ULMFiT model for text classification tasks. Despite facing challenges during the deployment process, particularly with operating system-specific issues, the project effectively showcases the integration of machine learning models into scalable, cloud-based services. The microservice's ability to classify news articles efficiently has the potential to enhance content management and personalization on news platforms, providing significant value to end users. Future work could explore the expansion of the service's capabilities, including sentiment analysis and the addition of more nuanced categories, to further improve its utility in the dynamic landscape of digital news.

10. REFERENCES

Zhang, X., Zhao, J., & LeCun, Y. (2015). Character-level convolutional networks for text classification. arXiv. https://arxiv.org/abs/1509.01626

Howard, J., & Ruder, S. (2018). Universal language model fine-tuning for text classification. In Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (ACL 2018). arXiv. https://arxiv.org/abs/1801.06146

Google Cloud. (n.d.). Google Cloud Run documentation. Google Cloud. https://cloud.google.com/run/docs

Google Cloud. (n.d.). Google Cloud SDK documentation. Google Cloud. https://cloud.google.com/sdk/docs/install

Docker Inc. (n.d.). Docker documentation. Docker Inc. https://docs.docker.com/

Howard, J., & Gugger, S. (2020). Fastai: A layered API for deep learning. arXiv. https://arxiv.org/abs/2002.04688

PyTorch. (n.d.). PyTorch documentation. PyTorch. https://pytorch.org/docs/stable/index.html

Pedregosa, F., et al. (2011). Scikit-learn: Machine learning in Python. Journal of Machine Learning Research, 12, 2825-2830. https://scikit-learn.org/stable/

AG NEWS News dataset. Retrieved from https://raw.githubusercontent.com/mhjabreel/CharCnn_Keras/master/data/ag_news_csv/train.csv