CE314

Web Scraper AI

Name: Leong Jia Juin, Benjamin

Kaplan ID: CT0365371

PRID: LEONG41406

Use Case

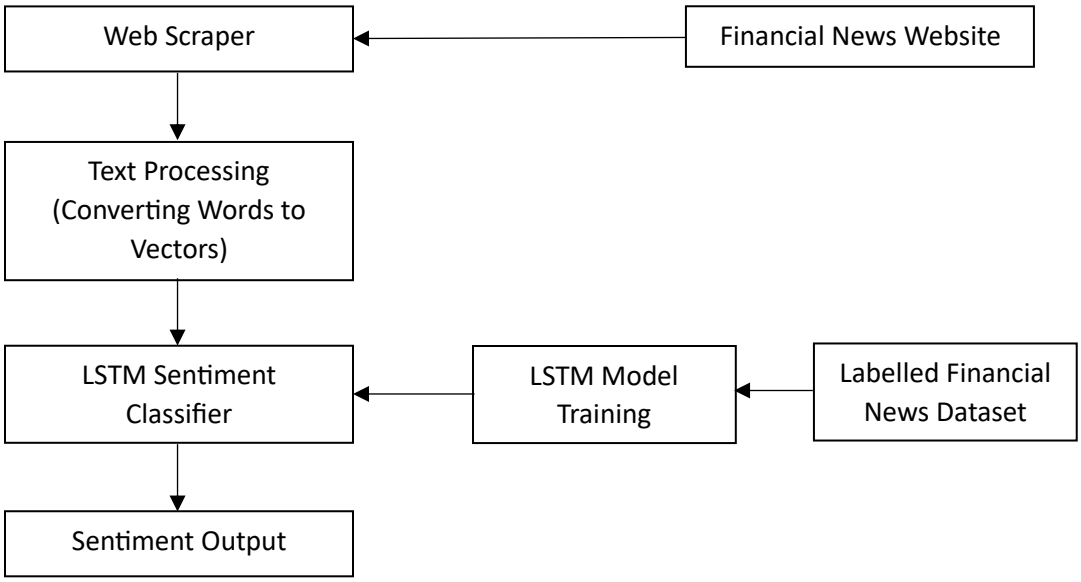
A possible use-case for a web-scraper AI would be a program that scrapes financial news headlines and performs sentiment analysis to uncover the underlying sentiments in the market. Financial markets are highly influenced by news and events. Making and accurate analysis of financial headlines is crucial for investors and traders. Machine learning techniques can extract valuable insights from vast amounts of textual data.

The program will first leverage web scraping techniques to collect financial headlines from various sources, such as news websites, financial blogs, and social media platforms. The headlines are then preprocessed such as to removing punctuation, stop words, and irrelevant information. Next, the headlines can be transformed into numerical representations, using natural language processing (NLP) techniques. Finally, a trained model will be employed to predict the sentiments of the financial headlines.

To capture the sentiments expressed in the headlines, the program employs a machine learning model trained specifically for sentiment analysis. Various machine learning models, such as support vector machines (SVM) or recurrent neural networks (RNN) can be employed for this task. The model is trained using a labeled dataset, where financial headlines are annotated with sentiment labels (e.g., positive, negative, or neutral). The program utilizes a combination of supervised learning and feature engineering to enhance the model's predictive accuracy. Once the sentiment analysis model is trained, it is applied to the collected financial headlines to classify their sentiments.

Overall, this web scraping AI can potentially be a powerful tool for investors to gain valuable insights into market sentiments, enabling them to make informed decisions based on sentiment trends and themes.

Proposed Wireframe



1. **Financial News Website:** This is the source from which the financial news headlines will be scraped from. A specific website that provide financial news will be chosen.
2. **Web Scraper:** This module will be responsible for scraping the headlines from the financial news website. It will fetch the HTML content, parse it, and extract the relevant headlines. The extracted headlines may contain noise or unwanted elements. This web scraper will perform necessary preprocessing tasks such as removing HTML tags, special characters, or stop words. It will also convert the text into a suitable format for the LSTM model.
3. **LSTM Sentiment Classifier:** An LSTM model is employed to classify the sentiment of each headline. It will take preprocessed headlines as input and output the sentiment classification.
4. **Sentiment Output:** This represents the final output of the system, where the sentiment (positive, negative, neutral) for each headline will be displayed or stored for further analysis.

Pseudo Code

Financial Dataset (Snapshot) - This is the dataset that will be used to train the LSTM Classifier

neutral	According to Gran , the company has no plans to move all production to Russia , although that is where the company is growing .
neutral	Technopolis plans to develop in stages an area of no less than 100,000 square meters in order to host companies working in compu
negative	The international electronic industry company Elcoteq has laid off tens of employees from its Tallinn facility ; contrary to earlier la
positive	With the new production plant the company would increase its capacity to meet the expected increase in demand and would impr
positive	According to the company 's updated strategy for the years 2009-2012 , Basware targets a long-term net sales growth in the range
positive	FINANCING OF ASPOCOMP 'S GROWTH Aspocomp is aggressively pursuing its growth strategy by increasingly focusing on technolo
positive	For the last quarter of 2010 , Componenta 's net sales doubled to EUR131m from EUR76m for the same period a year earlier , whil
positive	In the third quarter of 2010 , net sales increased by 5.2 % to EUR 205.5 mn , and operating profit by 34.9 % to EUR 23.5 mn .
positive	Operating profit rose to EUR 13.1 mn from EUR 8.7 mn in the corresponding period in 2007 representing 7.7 % of net sales .
positive	Operating profit totalled EUR 21.1 mn , up from EUR 18.6 mn in 2007 , representing 9.7 % of net sales .
positive	TeliaSonera TLSN said the offer is in line with its strategy to increase its ownership in core business holdings and would strengthen
positive	STORA ENSO , NORSKE SKOG , M-REAL , UPM-KYMMENE Credit Suisse First Boston (CFSB) raised the fair value for shares in four o
positive	A purchase agreement for 7,200 tons of gasoline with delivery at the Hamina terminal , Finland , was signed with Neste Oil OYj at th
positive	Finnish Talentum reports its operating profit increased to EUR 20.5 mn in 2005 from EUR 9.3 mn in 2004 , and net sales totaled EUI
positive	Clothing retail chain Sepp+Æl+Æ 's sales increased by 8 % to EUR 155.2 mn , and operating profit rose to EUR 31.1 mn from EUR 17
positive	Consolidated net sales increased 16 % to reach EUR74 .8 m , while operating profit amounted to EUR0 .9 m compared to a loss of
positive	Foundries division reports its sales increased by 9.7 % to EUR 63.1 mn from EUR 57.5 mn in the corresponding period in 2006 , and
positive	HELSINKI (AFX) - Shares closed higher , led by Nokia after it announced plans to team up with Sanyo to manufacture 3G handsets ,
positive	Incap Contract Manufacturing Services Pvt Ltd , a subsidiary of Incap Corporation of Finland , plans to double its revenues by 2007-
positive	Its board of directors will propose a dividend of EUR0 .12 per share for 2010 , up from the EUR0 .08 per share paid in 2009 .
positive	Lifetree was founded in 2000 , and its revenues have risen on an average by 40 % with margins in late 30s .

LSTM Model Training

1. **Import Dependencies**

```
import numpy as np
import pandas as pd
…
from keras.layers import LSTM, Dense, Embedding, Dropout
from keras.preprocessing.text import Tokenizer
from keras_preprocessing.sequence import pad_sequences
```

2. **Load Financial Dataset**

```
df = pd.read_csv('Financial_News_Dataset.csv', encoding = "ISO-8859-1")
```

3. **Dataset Preprocessing**

```
df['news'] = df['news'].apply(str.lower)
tokenizer = Tokenizer(num_words=5000, split=" ")
tokenizer.fit_on_texts(df['news'].values)
X = pad_sequences(X)
```

4. **Train Test Split**

```
X_train, X_test, y_train, y_test = train_test_split(X,y, test_size=0.2)
```

5. **Build LSTM Model**

```
model = Sequential()
model.add(Embedding(5000, 256, input_length = X.shape[1]))
```

6. **Train the Model**
```
model.fit(X_train, y_train, epochs = 10, batch_size = 32, verbose=2)
model.save(r'SentimentModel.h5')
```

7. **Test the Model**
```
model = load_model(r'SentimentModel.h5')
predictions = model.predict(X_test)
```

Web Scraper

1. **Import Dependencies**
```
from urllib.request import urlopen, Request
from bs4 import BeautifulSoup
import pandas as pd
```

2. **Load LSTM model**
```
model = load_model(r'SentimentModel.h5')
```

3. **Define website to scrape from**
```
finviz_url = 'https://finviz.com/quote.ashx?t='
tickers = ['AAPL']
```

4. **Scrape news headlines**
```
news_tables = {}
for ticker in tickers:
    url = finviz_url + ticker
    req = Request(url=url, headers={'user-agent': 'my-app'})
    response = urlopen(req)
    html = BeautifulSoup(response, features='html.parser')
```

5. **Parse headlines into dataframe**
```
parsed_data = []
for ticker, news_table in news_tables.items():
    for row in news_table.findAll('tr'):
        title = row.a.text
```
6. **Process text to numerical vectors**
```
df = df['title'].apply(str.lower)
df = df.apply(lambda x: re.sub('[^a-zA-Z0-9\s]',"",x))
tokenizer = Tokenizer(num_words=5000, split=" ")
tokenizer.fit_on_texts(df.values)
```

7. **Pad the vectors to fit the model**
```
padding_size = 50 - len(X[0])
padding_vector = np.zeros(padding_size)
```

8. **Convert prediction back to text format**
```
def format_predictions(predictions)
```

9. **Output a .csv file**
```
combined_df.to_csv('sentiments.csv', index=False)
```

Functional Code

1. Run 'SentimentAnalysis_TrainModel.py' to train the model (Please note that model has been trained and saved under the file 'SentimentModel.h5')
2. Run 'WebScraper.py' to scrape financial news headlines and analyze sentiments

```
16  #Define the website to scrape from
17  finviz_url = 'https://finviz.com/quote.ashx?t='
18  tickers = ['AAPL']
19
```

You may edit the tickers to scrape your choice of stock news

3. The output of WebScraper.py is a .csv file with the headlines and sentiments.