Movie Review Analysis Based on Sentiment Using Text-Processing API

Hasith Dissanayake

Table of Contents

1 Introduction

1.1 Overview

Movie reviews are one of the ways which can be used to determine how well the movie has performed. Even though numerical values gives a surface idea as to how well made the movie is, in order get a deeper understanding, movie reviews from critics and audience have to be considered. Sentiment analysis helps in this regard as it can be used to recognize the intention conveyed through text. This is used by businesses when taking feedback and it helps to understand the customer base and provide better products and services.

In this scenario using sentiment analysis a solution is developed to see whether the audience reaction to a particular is positive, negative, or neutral. In doing so, without going through the entire review it is possible to understand whether the viewer liked the movie or not.

1.2 Purpose

The purpose of this project is to build a web application which analyses movie reviews. Upon analysis the application will output whether the review is positive, negative or neutral.

In order to build this application the use of API and its integration will be understood. Knowledge of Flask which is a Python web frame will also be gathered as it used to develop the application.

2 Literature Survey

2.1 Existing problem

Investigations into how critics affect box office revenue have explored how critics have played both the predictor and the influencer roles. During the study which had been done over an eight-week period, it has been found that both positive and negative reviews have corresponded with weekly box office revenue. Impact of negative reviews have been more prominent when compared with the impact of positive reviews during the first week, suggesting evidence of a negativity bias. However, the impact from negative reviews seems to diminish over time when compared with the impact from positive reviews (Basuroy, et al., 2003).

When reviews taken into regard individually, is is found that some critics have been found to be incrementally influential and they are most likely to be pampered by producers. Further studies have a possibility reveal that specific critics might have an influence on a specific demographic for an example, influencing the youth (Boatwright, et al., 2007).

As previously noted, research has been conducted to investigate what past studies have sought to explain regarding the influence of film reviews and how they affect both consumers and product sales.Thorough this literature survey the author has been able to recognize how the information that the review provides positive, negative or mixed reactions which might affect consumer decision making process.

It is apparent that there is a need for the analysis of reviews in large scale, however the the amount of data is so overwhelming that it is difficult for a human being to analyse. This is where sentiment analysis comes in. It is used to differentiate viewers based on their attitudes towards the movie. It can be found whether the product review is positive or negative. There are two methods of performing sentiment analysis which are knowledge based and machine learning. The latter is found to be the simpler method (Teja, et al., 2018). Thorough this survey it has been found that sentiment analysis methods is one of the solution the issue of analysing movie reviews.

2.2 Proposed solution

The proposed solution which will be developed through this project for the problem of analysing movie reviews is a web application based on sentiment analysis. As previously discussed sentiment analysis through machine learning is one of the more efficient ways of analysing a opinions of reviewers. Therefore it will be used in the development of this project. The operation of analysis will be outsourced to another service through a text-processing API.

3 Theoretical analysis

3.1 Block diagram

The following figure is a diagrammatic overview of the project.

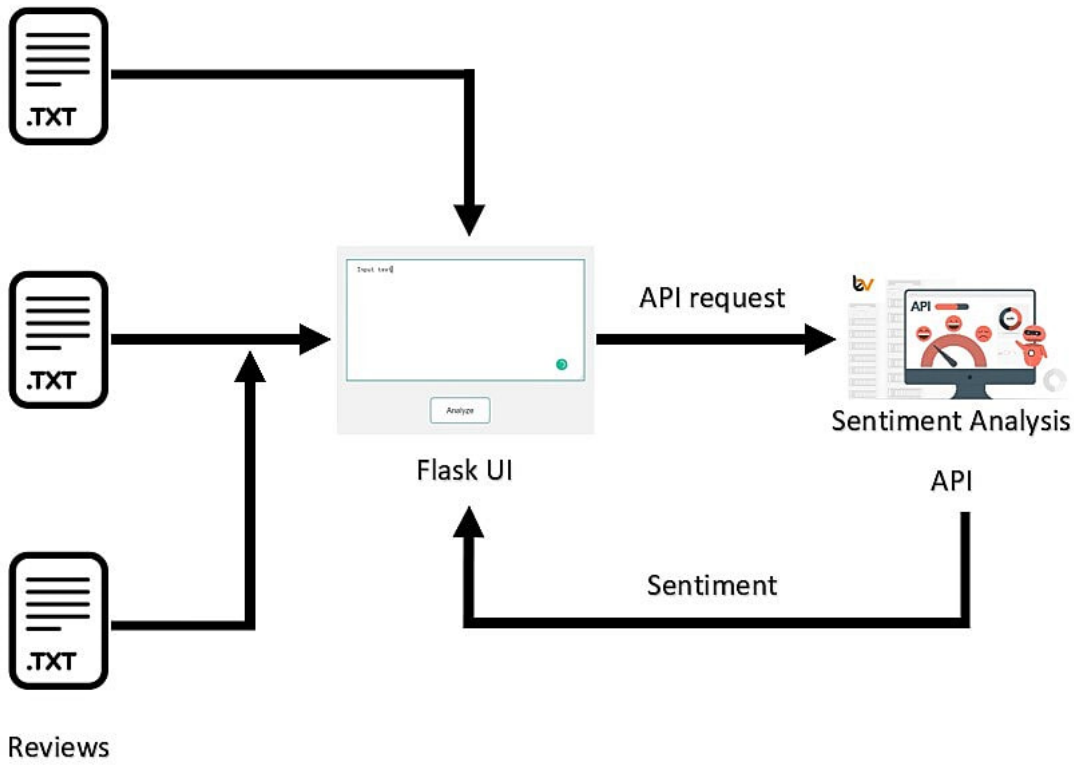

Figure 1: Block diagram

3.2 Software designing

The web application include the following languages and frameworks.

- Python 3.9.6
- flask 2.0.2

In order to run the application the minimum system system requirements are as follows.

- Modern Operating System:
    - Windows 7 or 10
    - Mac OS X 10.11 or higher, 64-bit
    - Linux: RHEL 6/7, 64-bit
- x86 64-bit CPU (Intel / AMD architecture)
- 4 GB RAM
- 5 GB free disk space

(March, 2020)

4 Experimental investigations

As the solution heavily relies on Flask framework, investigations were done to find the relevant libraries to import. Information regarding these were found in the API section in the documentation.

An application oriented API was also required to be included and upon investigation a Text-Processing API from japerk was found to be the suitable and it was most appealing due to its free price plan.

5 Flowchart

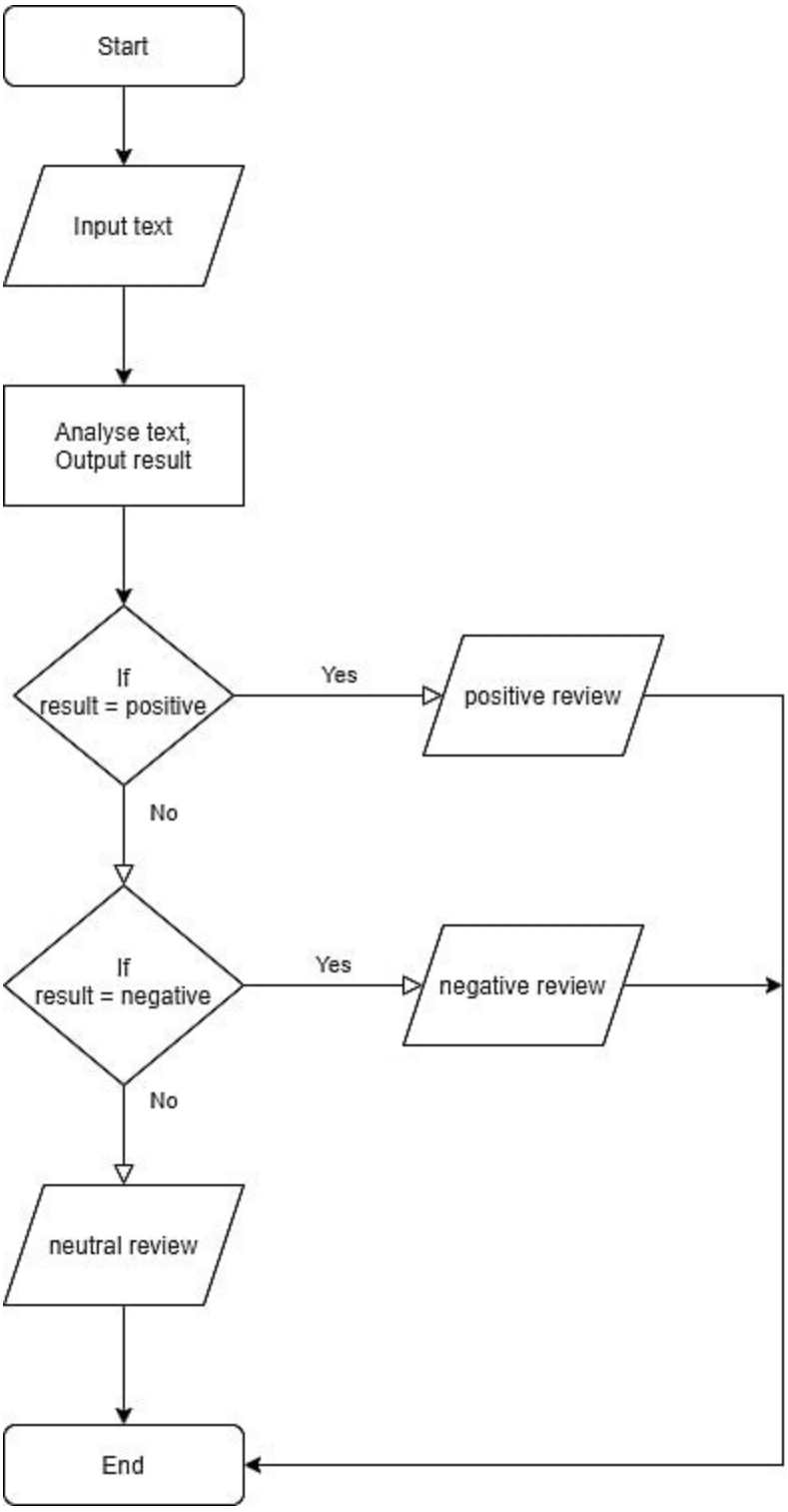

Figure 2: Flowchart of the application

6 Result

The result for putting a positive review such as "The movie was great." is shown below.

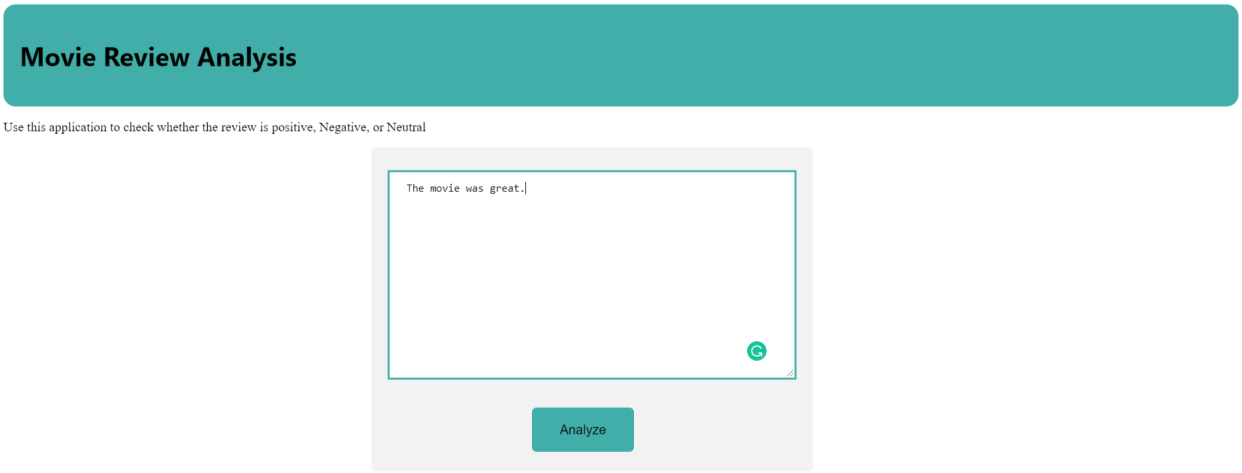

Figure 3: Inputting positive review

Figure 4: Result of positive review

The result of putting a negative review such as "The movie was bad" is shown below.

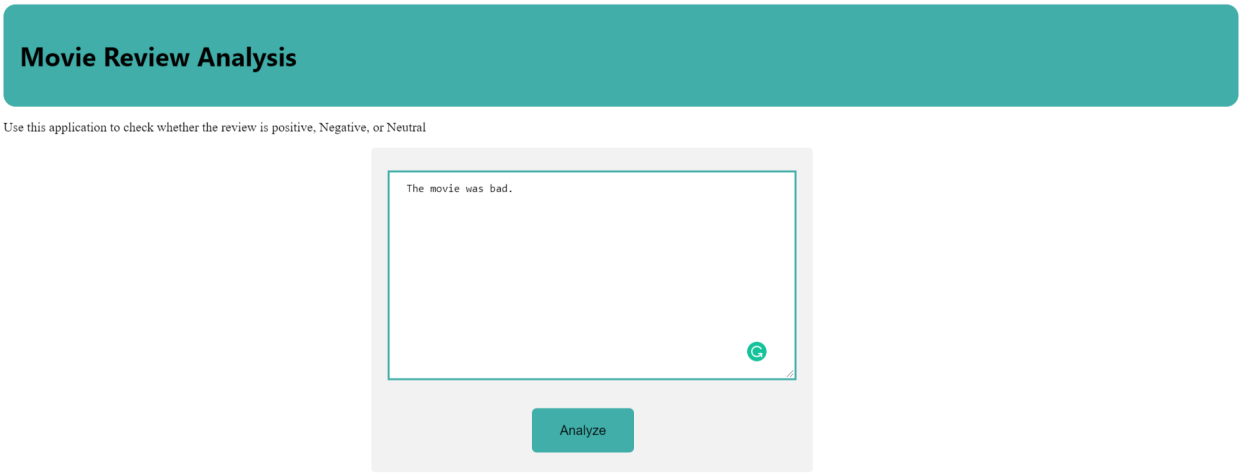

Figure 5: Inputting negative review

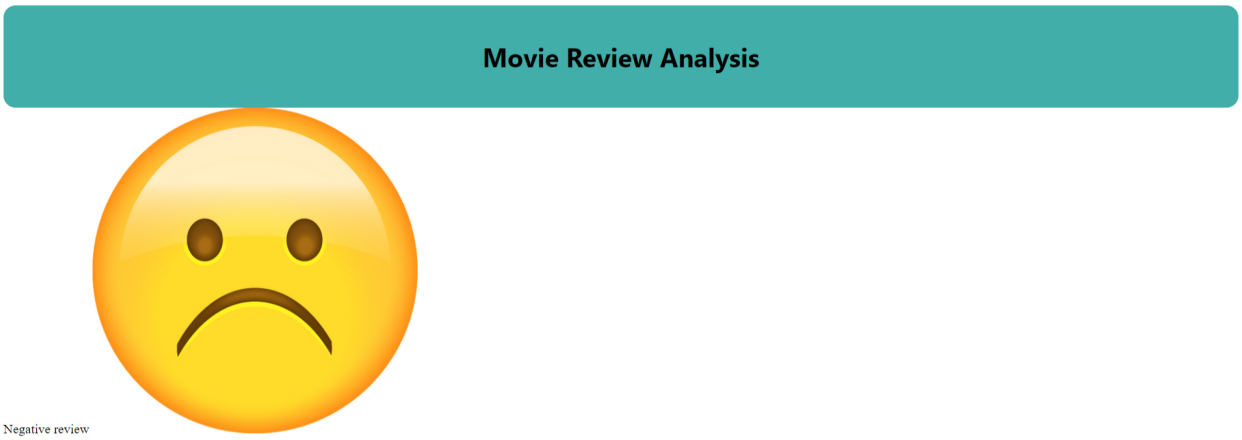

Figure 6: Result of negative review

The result of putting a neutral review such as "Did not feel anything." is shown below.

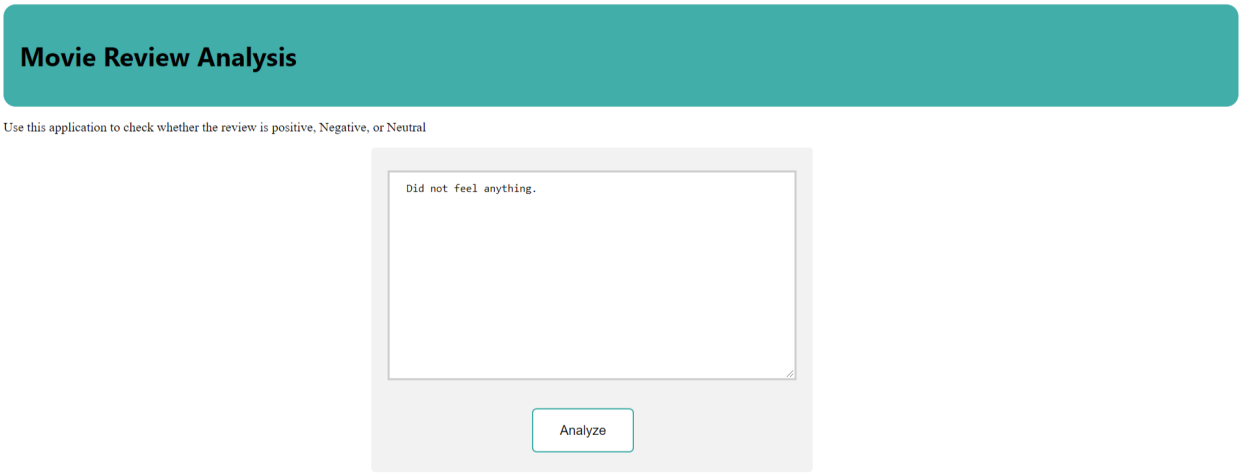

Figure 7: Inputting a neutral review

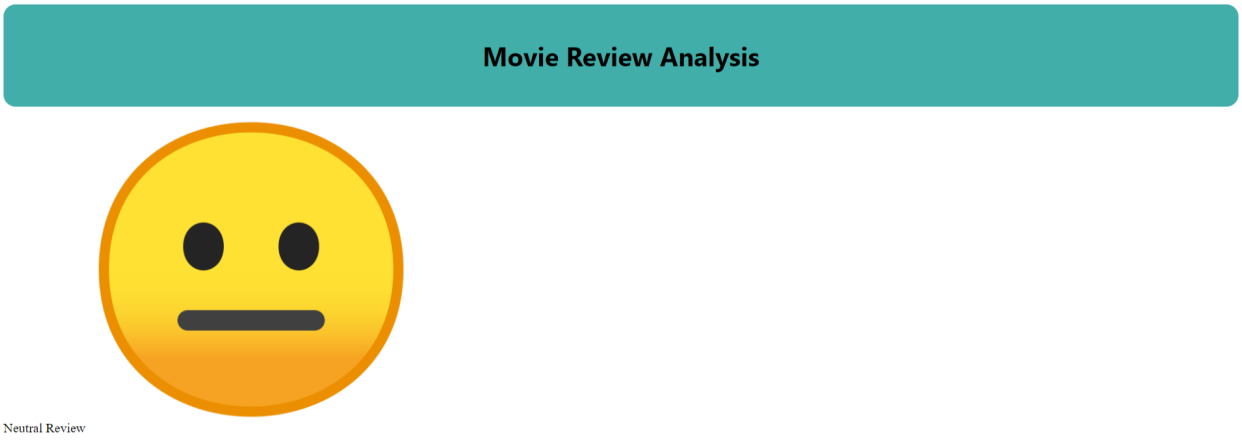

Figure 8: Result of neutral review

7 Advantages and disadvantages

Advantages

- Reviews can be analysed without having to read the entire text.
- Saves a considerable amount of time.
- Depending on the API plan an upwards of 8000 words can be analysed at a time.
- Intuitive UI.
- Instant results.

Disadvantages

- Service might be unavailable if the text processing server goes down.
- Text-processing algorithm still cannot clearly differentiate between a negative and a neutral review.

8 Applications

This application is specifically designed for analysing movie reviews. Therefore movie studios can make use of this application to analyse data and understand opinions of thousands of viewers thus taking necessary decisions to appeal to a wider audience and increase box office revenue.

However, as this application uses sentiment analysis and text-processing it can also be employed in other areas such as,

- Social media monitoring
- Brand monitoring and reputation management
- Customer support and feedback
- Product analysis
- Voice of customer and voice of employee
- Market research and competitive research

(Roldós, 2020)

9 Conclusion

In this project the author has discussed why movie review analysis is important and how movie studios can benefit from analysing reviews. The method of analysing text using sentiment analysis was heavily discussed and has been taken as the basis for the developed application. The author was required to gain knowledge on the use of API and the Flask framework for Python in order to develop the application and integration of a Text-Processing API was also done. The demonstration and the results of the application was shown to be successful. However there is room for improvement in the application in future releases.

10 Future Scope

One of the disadvantages which was found in this application is that the text processing algorithm has a hard time differentiating between a negative and a positive review. Therefore in the future a more reliable and accurate text processing API from Google Analytics or Microsoft Text analytics can be integrated. From a front-end perspective the UI can be also be made to be more dynamic and aesthetically pleasing.

11 Bibliography

- Basuroy, S., Chatterjee, S. & Ravid, S. A., 2003. How Critical Are Critical Reviews? The Box Office Effects of Film Critics, Star Power, and. *Journal of Marketing*, 67(4), pp. 103-117.
- Boatwright, P., Basuroy, S. & Kamakura, W., 2007. Reviewing the reviewers: The impact of individual film. *Quantitative Marketing and Economics*, 2007(5), pp. 401-425.
- March, J., 2020. *Enthought Python Minimum Hardware Requirements*. [Online] Available at: https://support.enthought.com/hc/en-us/articles/204273874-Enthought-Python-Minimum-Hardware-Requirements [Accessed 11 1 2022].
- Roldós, I., 2020. *8 Applications of Sentiment Analysis*. [Online] Available at: https://monkeylearn.com/blog/sentiment-analysis-applications/ [Accessed 11 1 2022].
- Teja, J. S., Sai, G. K. & Manikandan, R., 2018. Sentiment Analysis of Movie Reviews Using Machine Learning. *International Journal of Pure and Applied Mathematics*, 118(20), pp. 3277-3284.

Appendix

A. Source code

A.1. App.py

```
from flask import Flask, request, render_template
import requests
import re
import json
import configparser
config =configparser.ConfigParser()
config.read("config.ini")
apiKey= config["DEFAULT"]["api-key"]

app = Flask(__name__)

def check(output):
    url = "https://japerk-text-processing.p.rapidapi.com/sentiment/"
    payload = {"text":output}
    print(payload)
    headers = {
        'content-type': "application/x-www-form-urlencoded",
        'x-rapidapi-host': "japerk-text-processing.p.rapidapi.com",
        'x-rapidapi-key': apiKey
        }

    response = requests.request("POST", url, data=payload,
headers=headers)
    print(response.text)
    value = response.text
    output = json.loads(value)
    return response.json()

@ app.route('/')
def summarizer():
```

```
    return render_template('summarizer.html')

@app.route('/summarize', methods=['POST'])
def summarize():
    output = request.form['output']
    output=re.sub("[^a-zA-Z.,]"," ",output)
    print(output)
    essay = check(output)
    print(type(essay['label']))
    if essay['label'] == "pos":
        output="Positive review"
    elif essay['label'] == "neg":
        output="Negative review"
    else:
        output="Neutral Review"
    return
render_template('summary.html',essay=essay,prediction_text='{}'.format(out
put))
if  __name__=="__main__":
    app.run()
```

A.2. Summarizer.html (CSS excluded)

```
<body>
  <div class="header">
    <h1>Movie Review Analysis</h1>
  </div>
  <div>
    <p>Use this application to check whether the review is positive,
Negative, or Neutral</p>
  </div>
  <div id="form">
    <form action="/summarize" method="post">
      <textarea class="textarea" name="output" rows="15" cols="60"
required></textarea>
```

```
      <br><br>
      <input class="submit" type="submit" value="Analyze">
    </form>
  </div>
</body>
```

A.3. Summary.html (CSS excluded)

```
<body>
    <div class="header">
        <h1>Movie Review Analysis</h1>
    </div>
    <div>
        {% if essay["label"]=="pos" %}
            {{prediction_text}}
            <img src="static/images/positive.png" alt="positive image"
width="400" height="400">
        {% elif essay["label"]=="neg" %}
            {{prediction_text}}
            <img src="static/images/negative.png" alt="positive image"
width="400" height="400">
        {% else %}
            {{prediction_text}}
            <img src="static/images/neutral.png" alt="positive image"
width="400" height="400">
        {% endif %}
    </div>
</body>
```