CAPSTONE PROJECT
MOVIE SPOILER DETECTION

TEAM: KANYARASHI

SQUAD:

AKASH GUJE
BHANU PRAKASH
NOAH DAVID
PRAMOD REDDY GURRALA
SAI VARUN KOLLIPARA

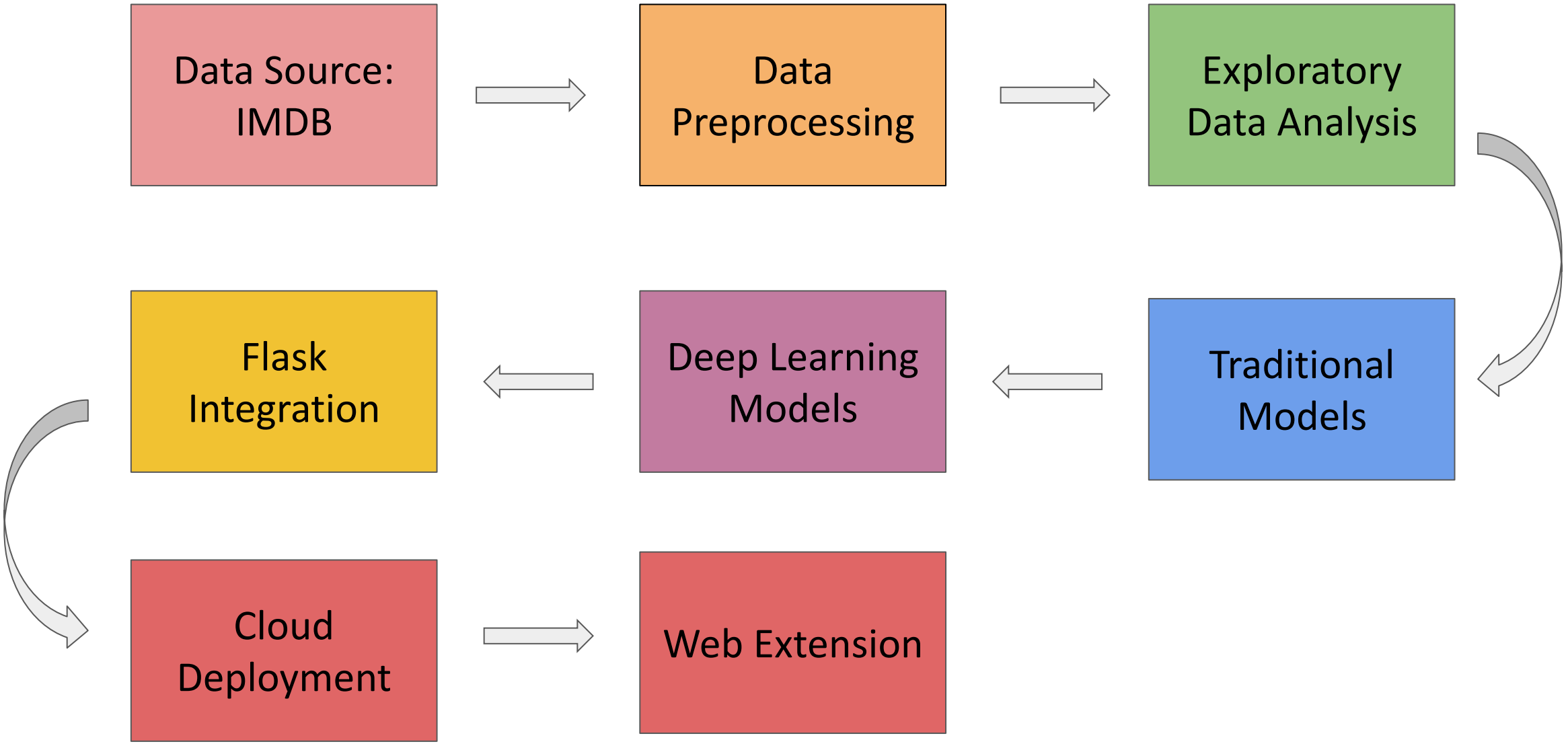
Introduction:

- Movie spoilers are a common problem for movie enthusiasts who want to enjoy a film without any prior knowledge of the plot. In this project, we propose a solution to predict the presence of spoilers in movie reviews using natural language processing techniques and blur them on any website so that it is not visible for the end users of that website

Motivation:

- The business problem that we aim to address is the presence of movie spoilers in online reviews. Spoilers not only ruin the experience for movie-goers but can also lead to a decrease in box office sales. Moreover, websites that allow spoilers can lose traffic and revenue due to the negative impact on user experience.

Methodology:

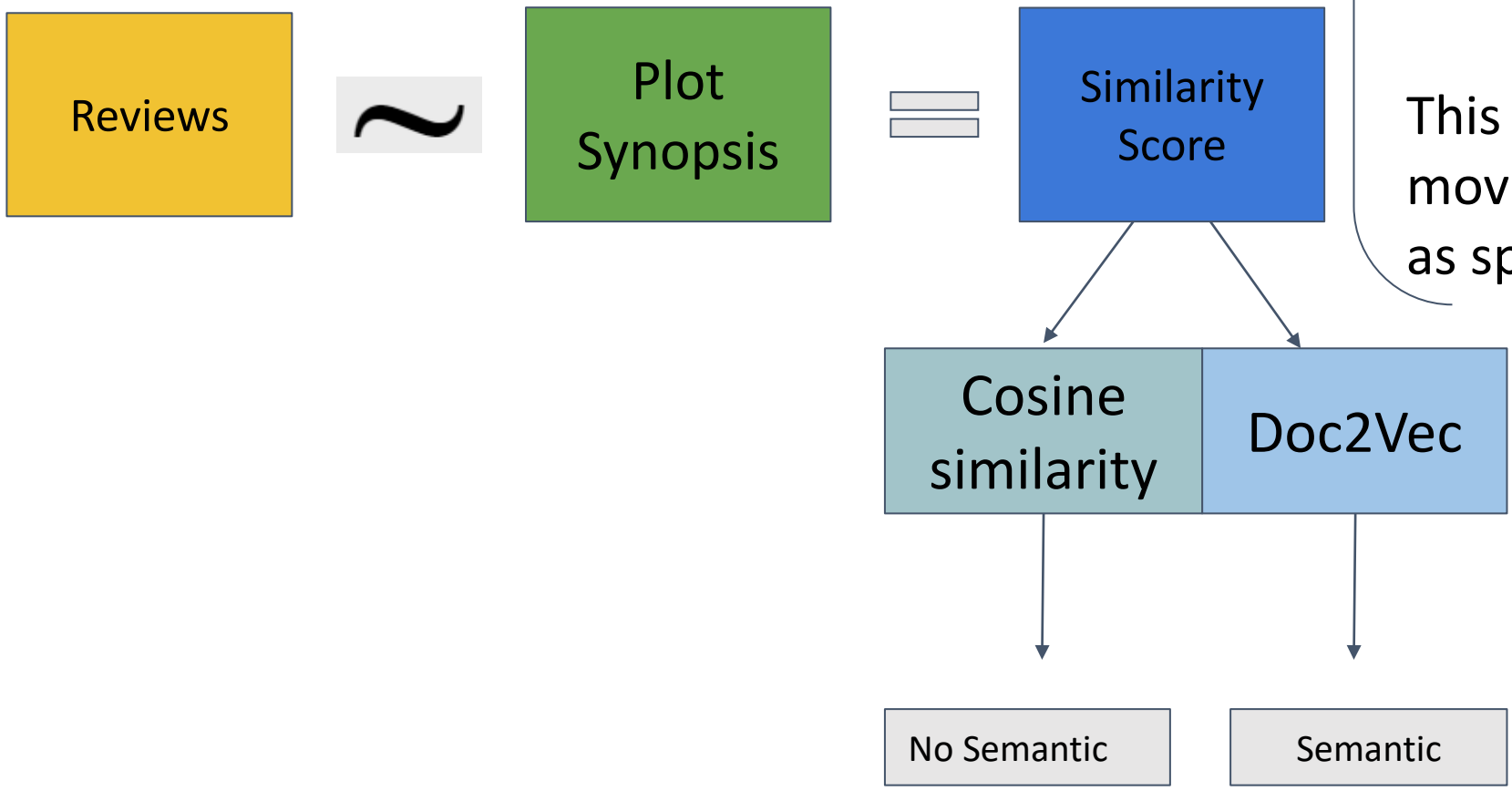


Data Collection

➔ To extract the reviews from IMDB website, we have leveraged python's Beautiful soup package to extract reviews from "https://www.imdb.com/chart/top/"

No.of Spoilers	6336
No.of Non Spoilers	16534

Exploratory Data Analysis



Similarity score gave us an insight on how each movies are classified us spoilers and non spoilers.

This also showed how some movies were missed classified as spoiler.

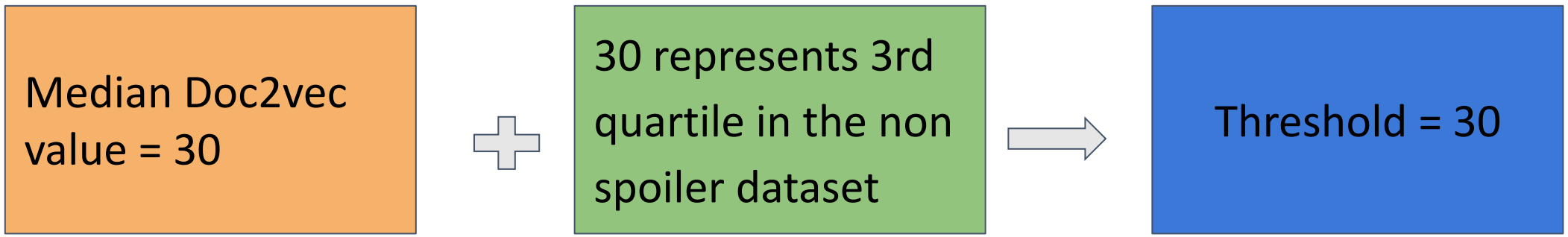
Exploratory Data Analysis - Continues

But can we consider similarity as a feature to be included in identifying a review as a spoiler or not?

To verify this we performed Hypothesis testing:

H0 : There is no significant difference	Mann-Whitney U Test (alpha - 0.05)	Result
Ha : There is a significant difference		**P-value: 0.0, hence H0 rejected**

From a user point of view and from a model point of view, It is important for the model to identify non spoilers as spoilers than classifying spoiler as non spoilers.



Exploratory Data Analysis - Continues

But can we consider 30 as threshold?

To verify this we performed Hypothesis testing:

H0: There is no relationship	Chi- squared test (alpha: 0.05)	Results
Ha: There is a relationship		**P-value = 0.0, hence H0 rejected**

Updated Dataset size:

Spoilers data	10216
Non Spoiler data	12663

Models

Traditional Modeling:

Model Training

Model	Count vectorizer		Tfidf Vectorizer	
	Recall	FNR	Recall	FNR
Logistic Regression	56%	28.8%	61%	27.5%
SGD Classifier	53%	27.2%	62%	27%
Random Forest Classifier	58%	31%	62.9%	27.5%

Model Testing

Model	Count vectorizer		Tfidf Vectorizer	
	Recall	FNR	Recall	FNR
Logistic Regression	60%	31.8%	63%	29%
SGD Classifier	57%	34%	63%	29%
Random Forest Classifier	58%	33%	65%	28%

Neural Networks

After implementing the traditional model, we tried exploring deep learning neural network starting with Artificial Neural network model with three fully connected layers and 512 neurons in each layer. After performing 20 epochs the following are the results:

Loss	Train Accuracy	Test Loss	Test accuracy	FNR
0.5709	0.7034	0.5854	0.7063	25%

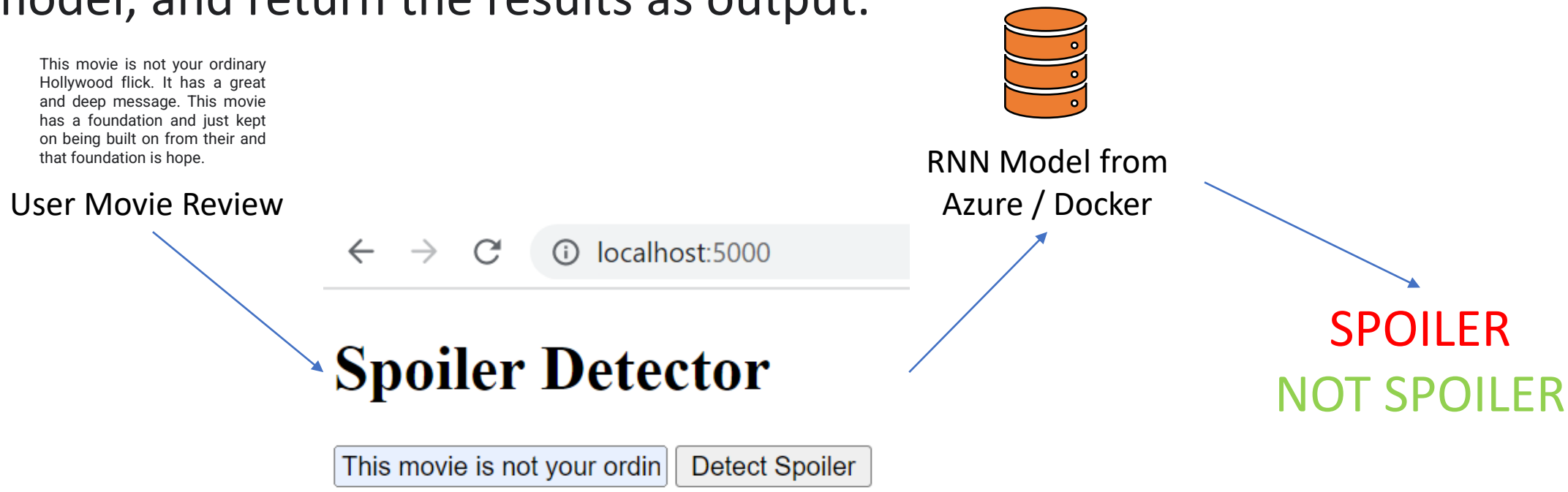
We also implemented LSTM:

Loss	Train Accuracy	Test Loss	Test accuracy	FNR
0.3742	0.8373	0.6962	0.6732	28%

The lstm model was completely overfitting. The DNN model has given a good result and the model is also stable hence we have decided to go ahead with this model's implementation for our final output as well.

Flask Integration

- Flask is a lightweight web framework that can be used to deploy machine learning models as web services. With Flask, you can create a RESTful API that can receive input data, run it through your ML model, and return the results as output.

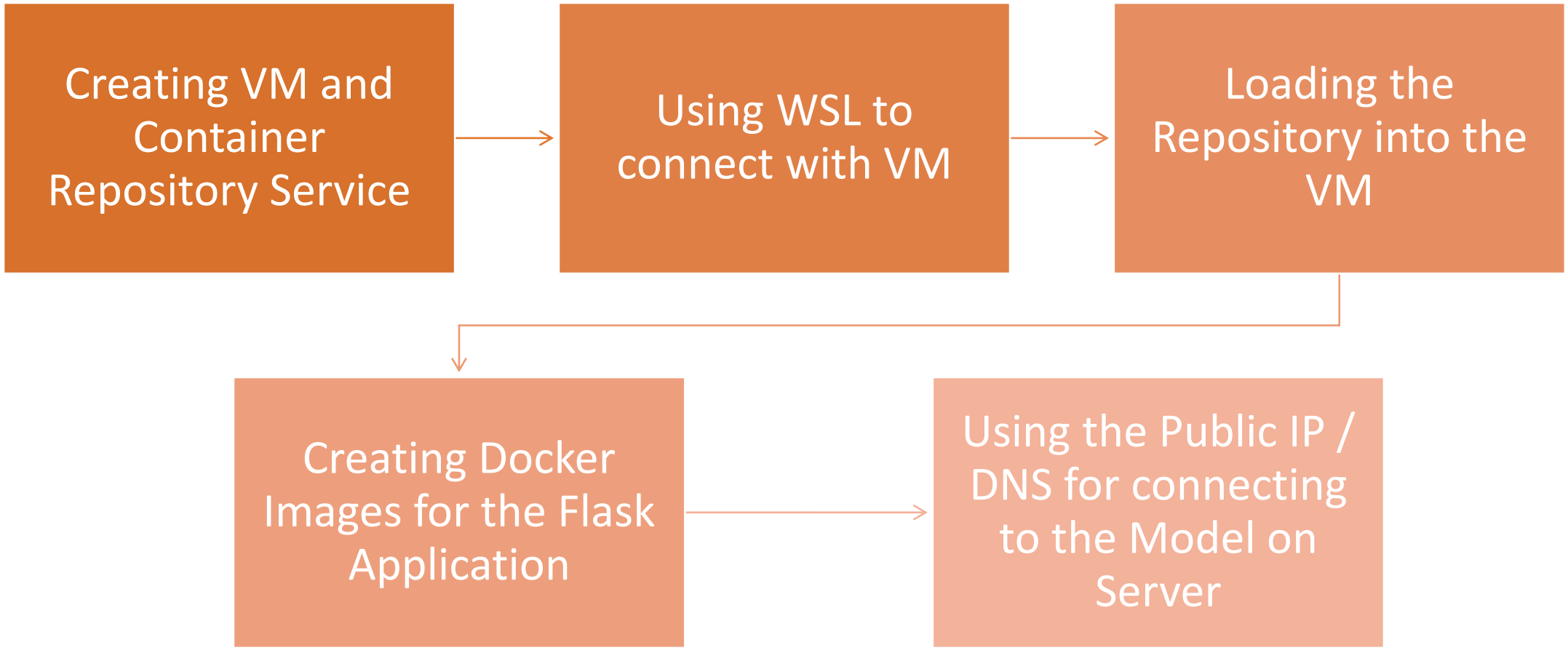


Web - Extension

- User friendly interface and easy to load on popular browsers
- Easy and fast to detect spoilers and hides the reviews from reading
- Real-Time Prediction

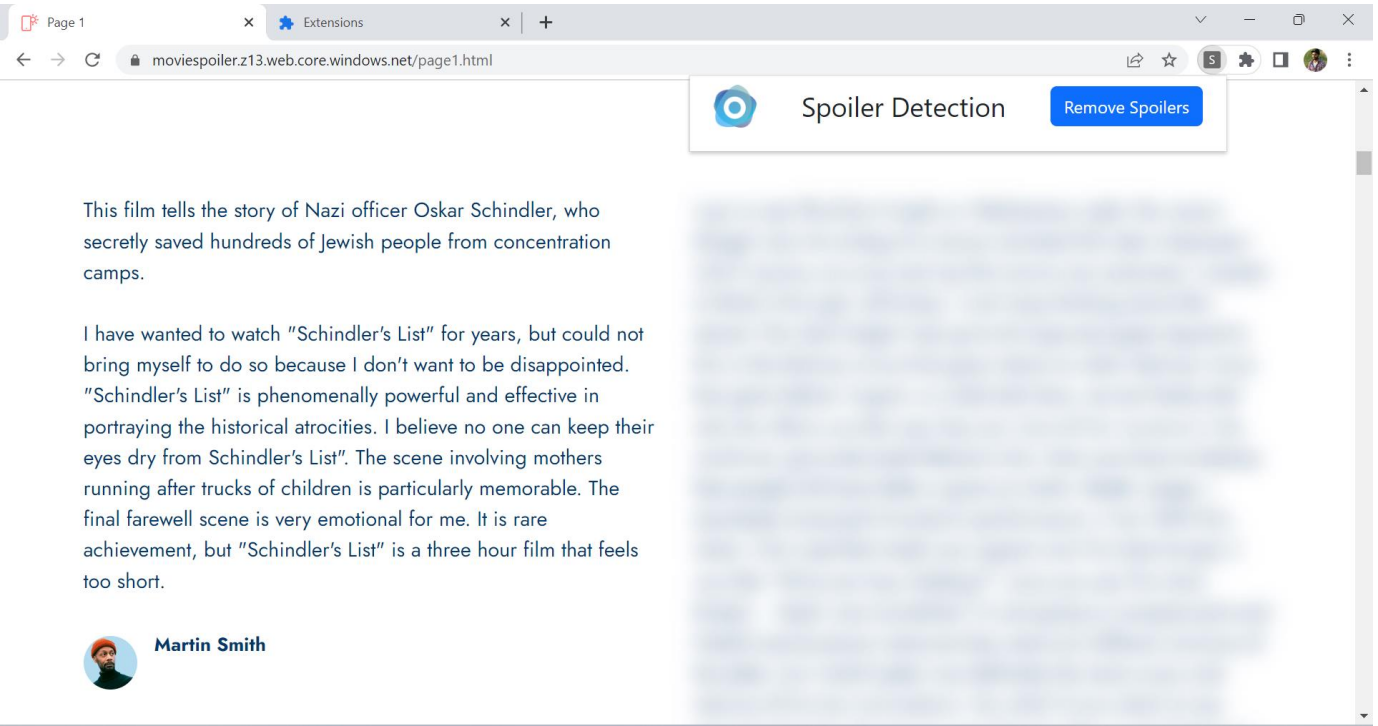
Preview of the Chrome Extension

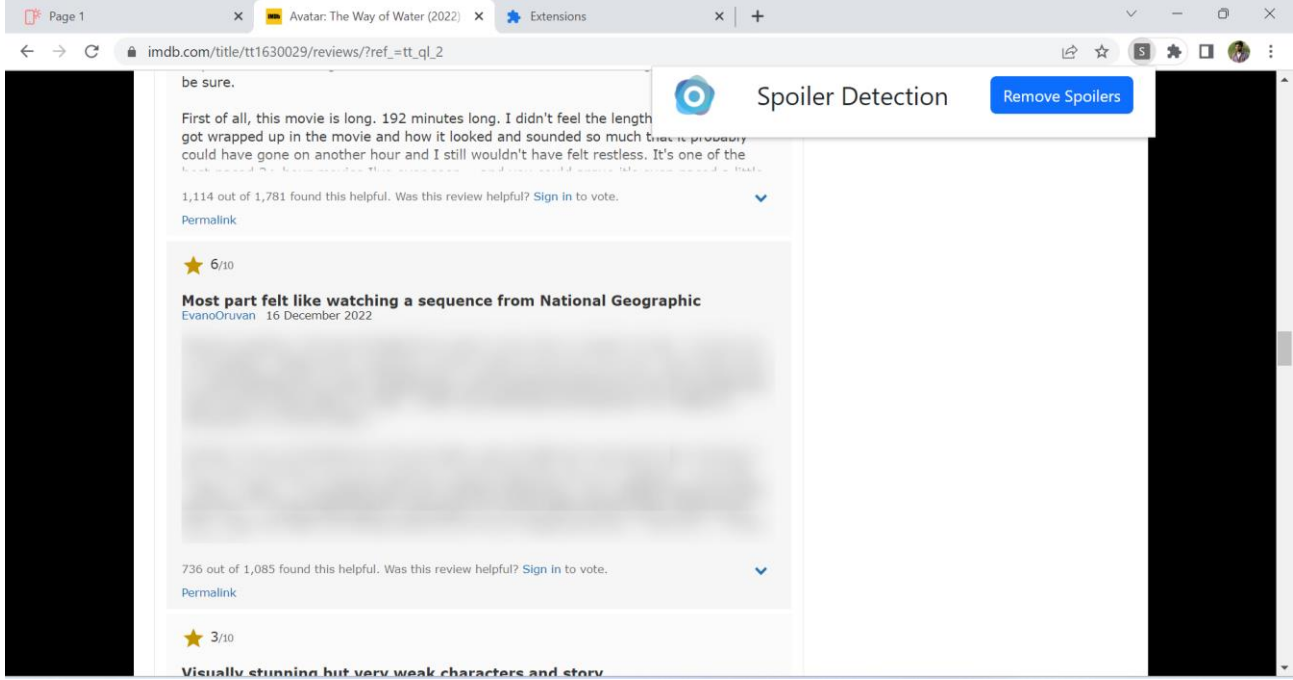
Microsoft Azure Hosting



Results

Sample Static Website Comments





IMDB Website User Reviews

Conclusion and Future Developments

- To conclude, we have created a google chrome web extension that can identify and blur the spoilers from movie review websites. We have built our own Neural Network models trained and tested with the review data we scrapped from multiple online resources.
- Furthermore, we have done Flask integration, Docker Deployment, and Azure Cloud Hosting with the model. We are limited to working on the Chrome browser and the IMDB website for this research.

- Future Developments:
  - Collaborate with movie studios
  - Expand to other platforms
  - Integration with social media platforms

THANK YOU