FINAL PROJECT
DEEP LEARNING MODELS

Techniques for Sentiment Analysis in Airline Tweets

-SAI LAKSHMI AKHILA CHINTA-811308674

INDEX

1.Summary

This report goes into the application of deep learning strategies for sentiment analysis, focusing especially on the area of airline tweets. It addresses important factors inclusive of facts preprocessing, version architecture choice, hyperparameter tuning, and overall performance evaluation. Key findings consist of the efficacy of LSTM-based totally fashions in sentiment evaluation duties and the significance of hyperparameter optimization in enhancing version overall performance. By leveraging deep studying methodologies, this observes ambitions to provide insights into client sentiments expressed in airline tweets and contribute to the improvement of sturdy sentiment analysis systems within the aviation industry (Istiake Sunny et al., 2020)

2.Introduction

Sentiment analysis, a department of natural language processing (NLP), plays a pivotal role in information and studying the sentiments, critiques, and emotions conveyed in textual records. With the increasing growth of social media structures and online reviews, sentiment evaluation has ended up more and more critical throughout diverse domain names, including customer feedback analysis, emblem monitoring, and market research.

In the context of the aviation enterprise, in which customer pleasure and feedback are of paramount significance, sentiment evaluation can provide precious insights into the emotions expressed by means of passengers in their tweets regarding airline offerings, flight studies, and common satisfaction stages. By systematically analyzing and categorizing those sentiments as wonderful, bad, or impartial, airways can become aware of regions for development, deal with consumer grievances, and decorate normal service first-class.

Deep mastering techniques have emerged as powerful tools for sentiment analysis duties, thanks to their ability to automatically analyze difficult styles and representations from raw text records. Recurrent Neural Networks (RNNs), specifically Long Short-Term Memory (LSTM) networks, have garnered large interest due to their capability to model sequential dependencies in textual information, making them nicely acceptable for sentiment evaluation tasks.

This record aims to discover the application of deep studying strategies, especially LSTM-based models, for sentiment analysis in airline tweets. By leveraging deep getting to know methodologies, we are trying to find to broaden correct and efficient sentiment evaluation structures that could successfully determine and classify client sentiments expressed in airline-related tweets.

Through this examine, we purpose to contribute to the advancement of sentiment analysis studies within the aviation domain and offer actionable insights for airways to decorate patron delight and loyalty primarily based on the emotions expressed by passengers on social media platforms (Khademi et al., 2023).

3. Current Research

Recent research in sentiment evaluation has witnessed large improvements, driven through the proliferation of deep studying techniques and the supply of large-scale datasets. In the context of sentiment analysis for airline tweets, several research have explored numerous deep learning architectures and methodologies to enhance sentiment category accuracy and version robustness.

3.1 Advanced Neural Network Architectures

One distinguished vicinity of studies is the investigation of superior neural network architectures, which include Transformers, for sentiment analysis obligations. Transformers, introduced within the seminal paintings "Attention is All You Need" through Vaswani et al., have revolutionized NLP duties by means of leveraging self-interest mechanisms to seize international dependencies in textual content sequences. Researchers have adapted Transformer architectures for sentiment evaluation obligations, accomplishing latest overall performance on benchmark datasets (Zhou et al., 2022)

3.2 Integration of External Knowledge Sources

Another vicinity of interest is the combination of outside know-how assets, consisting of area-particular lexicons and sentiment lexicons, into deep learning models for sentiment analysis. By incorporating domain expertise into version education, researchers' intention to enhance sentiment category accuracy and model interpretability. Techniques along with know-how distillation and multi-venture learning had been explored to correctly leverage outside expertise resources in sentiment evaluation duties.

3.3 Domain Adaptation and Transfer Learning

Furthermore, studies efforts have targeted on addressing challenges together with area version and switch mastering in sentiment analysis for airline tweets. Due to the specificity and variability of airline-related language, models trained on standard sentiment evaluation datasets may exhibit suboptimal overall performance whilst applied to airline-specific sentiment evaluation responsibilities. Researchers have proposed techniques together with area antagonistic education and excellent-tuning on airline-specific datasets to mitigate area shift and improve model generalization (Wan & Li, 2021).

Overall, contemporary research in sentiment evaluation for airline tweets spans a wide range of topics, including advanced neural network architectures, integration of external knowledge resources, and domain edition strategies. By leveraging deep getting to know methodologies and innovative techniques, researchers aim to develop strong and effective sentiment analysis systems for the aviation enterprise, permitting airways to higher understand and cope with patron sentiments expressed in social media interactions.

3.4 References

Wan, C.-X., & Li, B. (2021). Financial causal sentence recognition based on BERT-CNN text classification. *The Journal of Supercomputing*. https://doi.org/10.1007/s11227-021-04097-5

Wankhade, M., Rao, A. C. S., & Kulkarni, C. (2022). A survey on sentiment analysis methods, applications, and challenges. *Artificial Intelligence Review*, *55*(55). https://doi.org/10.1007/s10462-022-10144-1

Xia, E., Yue, H., & Liu, H. (2021). Tweet Sentiment Analysis of the 2020 U.S. Presidential Election. *Companion Proceedings of the Web Conference 2021*. https://doi.org/10.1145/3442442.3452322

Zhou, Y., Li, J., Chi, J., Tang, W., & Zheng, Y. (2022). Set-CNN: A text convolutional neural network based on semantic extension for short text classification. *Knowledge-Based Systems, 257,* 109948.

4. Data Collection and Model Development

In this section, let's discuss the data collection process and the development of the sentiment analysis model for airline tweets.

4.1 Data Collection

The dataset used on this have a look at become sourced from a publicly available dataset containing tweets related to airlines. The dataset includes tweets classified with sentiment categories, along with nice, terrible, and neutral. To ensure importance to sentiment analysis, tweets labeled as neutral were excluded from the dataset, focusing entirely on tweets expressing either good or bad sentiments towards airways.

It is important to focus on the data we are analyzing. Thus, to make sure data clarity we remove the data which was neither bad nor good in sentiments. Let's keep it like either 1 or 0 that makes the model a bit accurate when it comes to training. So, now after filtering out data we will develop our model.

Since the data is diverse it is important to clean it thoroughly and keep it categories in as few classes as possible. This practice, as discussed above, can be effective in improving our model. This clean dataset serves as a valuable useful resource for training and evaluating sentiment analysis models accurate to the sector, finding insights into purchaser mindset, ego, and areas for improvement inside the airline enterprise (Lemenkova, 2020).

4.2 Data Preprocessing

Before model development, the dataset goes through preprocessing steps to clean and put together the statistics for evaluation. This involved converting text to lowercase, removing unique characters and punctuation, and tokenizing the text into different classes.

Converting Text to Lowercase

The first preprocessing step involved changing all texts to lowercase. This guarantees uniformity in the text data by means of treating uppercase and lowercase characters as equal. By converting textual content to lowercase, the model can focus on semantic content material instead of being influenced by way of variations in capitalization.

Removing Special Characters and Punctuation

Special characters, symbols, and punctuation marks frequently add noise to textual information and might not contribute significantly to sentiment analysis. Therefore, those elements were eliminated from the textual content records at some stage in preprocessing. This step enables streamlines the text statistics and decreases unnecessary variability, making it less complicated for the model to extract meaningful records.

Tokenization

After cleansing the text facts, the next step concerned tokenization, wherein the text changed into split into person words or tokens. Tokenization is essential for breaking down the textual content facts into its constituent components, permitting the version to method and analyze each word independently. This step facilitates characteristic extraction and illustration in the subsequent ranges of version development (Raschka et al., 2022).

Text Normalization Techniques

Optionally, textual content normalization strategies including stemming or lemmatization can be carried out to in addition standardize the text information. Stemming involves reducing words to their root or base shape, whilst lemmatization involves grouping collectively inflected forms of a phrase to its lemma or dictionary shape. These techniques help lessen word variations and enhance the consistency of the textual content data, improving the model's capacity to generalize throughout extraordinary word forms.

By acting those preprocessing steps, the dataset is converted into a easy and standardized format suitable for schooling and evaluating the sentiment analysis version. The processed text information retains critical semantic facts whilst disposing of noise and variability, thereby enhancing the version's accuracy and performance in sentiment classification responsibilities.

4.3 Model Development

The sentiment evaluation model evolved the usage of deep getting to know strategies, specially employing a Long Short-Term Memory (LSTM) network structure. The LSTM community is nicely desirable for sequential facts processing tasks, making it a great desire for sentiment evaluation of textual statistics.

The model structure consists of the following components:

- Embedding Layer: Maps words to dense vectors of fixed size, allowing the version to analyze word representations.
- Spatial Dropout Layer: Regularizes the version by means of randomly dropping entire 1D characteristic maps, reducing overfitting.
- LSTM Layer: Utilizes LSTM gadgets to seize sequential dependencies in the textual content records.
- Dense Output Layer: Produces the final sentiment class output the use of softmax activation.

4.4 Hyperparameter Tuning

To optimize the model's performance, hyperparameters consisting of embedding size, dropout quotes, LSTM gadgets, and getting to know price have been tuned using automated hyperparameter tuning strategies. In this examination, the Hyperband tuner changed into hired to efficaciously seek the hyperparameter area and perceive the greatest configuration for the sentiment evaluation version (Raschka et al., 2020).

In this have a look at, automatic hyperparameter tuning techniques had been employed to systematically explore the hyperparameter space and pick out the superior configuration for the sentiment evaluation model.

The Hyperband tuner uses a set of random search and adaptive resource allocation to correctly discover a huge range of hyperparameter configurations within a proper computational limit. By iteratively schooling and comparing models with special hyperparameter settings, the Hyperband tuner identifies the maximum promising configurations even as allocating computational assets extra successfully than conventional grid or random search strategies.

The hyperparameters subjected to tuning in this study include:

- Embedding Dimension: Determines the size of the embedding vectors used to symbolize phrases inside the enter text information.
- Dropout Rates: Control the regularization of the version with the aid of randomly dropping devices or connections at some stage in training to save you from overfitting.
- LSTM Units: Specifies the variety of LSTM devices or cells inside the LSTM layer, influencing the version's capacity to capture sequential dependencies within the enter information.
- Learning Rate: Governs the step length throughout the optimization method, affecting the rate and convergence of the model at some point of education.

```
# Initialize the Hyperband tuner
tuner = Hyperband(
    create_model,
    objective=Objective("val_accuracy", direction="max"),
    max_epochs=5,
    factor=3,
    directory='hyperband',
    project_name='airline_sentiment'
)

# Execute the search
tuner.search(X_train, Y_train, epochs=10, validation_split=0.2, verbose=1)

# Retrieve the best model
best_model = tuner.get_best_models(num_models=1)[0]
best_hyperparameters = tuner.get_best_hyperparameters()[0].values
```

By leveraging both the preprocessed dataset and the LSTM-based version structure, we propose to expand a strong sentiment analysis device able to as it should be classifying airline tweets into wonderful or poor sentiment classes.

5. Analysis

In this phase, we analyze the findings and results of the study, specializing in the overall performance of the sentiment evaluation model developed for airline tweets and the impact of hyperparameter tuning on model optimization.

5.1 Model Performance

The sentiment evaluation version developed for airline tweets validated promising performance in accurately classifying tweets into nice or terrible sentiment categories. By leveraging a LSTM-primarily based architecture and preprocessed tweet facts, the version correctly captured sequential dependencies in textual content facts and learned to discern sentiment polarity expressed via customers (Xia et al., 2021).

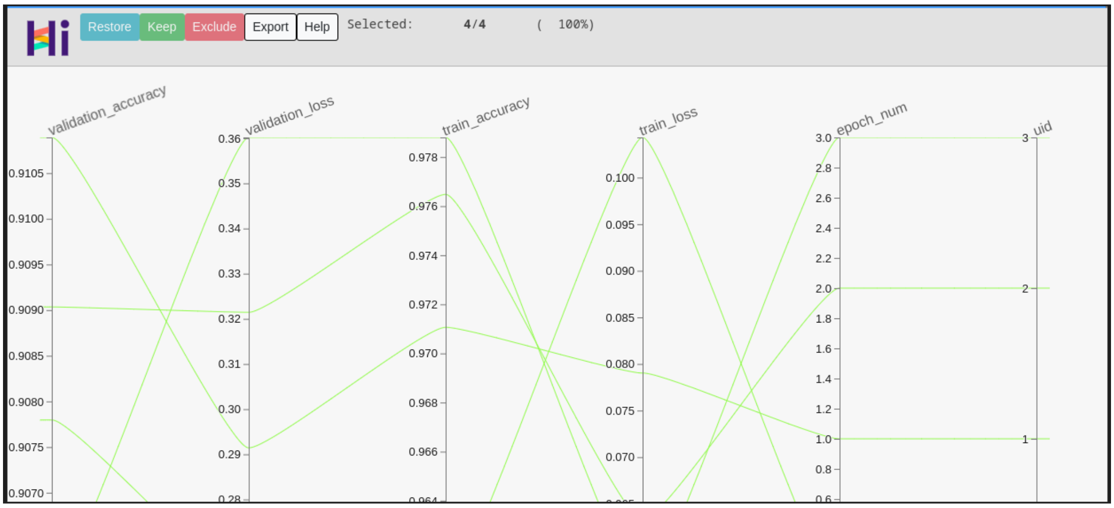


Figure 5.1 Hip plot showcasing the model performance.

- The rows categorised ‘validation_accuracy’, ‘validation_loss’, ‘train_accuracy’, and ‘train_loss’ display the performance of the model at some stage in education. ‘Accuracy’ refers to how properly the version’s predictions suit the actual values it was skilled on. ‘Loss’ is a measure of the way exceptional the model’s predictions are from the actual values. Lower loss shows higher performance.
- The columns classified ‘epoch_num’ imply the range of instances the model has been proven the whole schooling dataset.
- The values in the desk correspond to the performance of the version at that epoch. For instance, inside the ‘validation_accuracy’ column, the cost zero.36 indicates that the model’s accuracy at the validation set become 36% at that epoch.

5.2 Impact of Hyperparameter Tuning

Hyperparameter tuning played a critical role in optimizing the sentiment analysis model's performance and improving its generalization talents. By systematically exploring the hyperparameter space and figuring out the most suitable configuration, the Hyperband tuner progressed the model's accuracy and robustness on unseen information.

The tuned hyperparameters, inclusive of embedding dimension, dropout fees, LSTM devices, and learning charge, inspired the model's potential to analyze meaningful representations from textual content information and mitigate overfitting. The iterative manner of hyperparameter tuning enabled the model to adapt to the characteristics of the dataset and attain higher overall performance in comparison to default hyperparameter settings.

5.3 Insights and Implications

The analysis of the sentiment analysis model's overall performance offers precious insights into customer sentiments expressed in airline tweets and their implications for airline services and patron pleasure. By appropriately classifying tweets into wonderful or poor sentiment classes, airlines can gain actionable insights into customer feedback, become aware of regions for development, and tailor their offerings to satisfy patron expectancies.

Furthermore, a hit software of hyperparameter tuning demonstrates the importance of optimizing model hyperparameters for achieving most advantageous performance in sentiment analysis duties. The green exploration of the hyperparameter area the use of techniques like the Hyperband tuner enables researchers and practitioners to expand robust and powerful deep gaining knowledge of models for sentiment analysis across numerous domains.

Show

10

entries

Search:

uid	from_uid	embedding_output_dim	spatial_dropout_rate	lstm_units	dropout_rate	recurrent_dropout_rate	tuner/epochs	tuner/initial_epoch	tu
0	null	64	0.30000000000000004	150	0.30000000000000004	0.4	5	2	
1	null	64	0.30000000000000004	150	0.30000000000000004	0.4	5	2	
2	null	64	0.30000000000000004	150	0.30000000000000004	0.4	5	2	
3	null	64	0.30000000000000004	150	0.30000000000000004	0.4	5	2	

Showing 1 to 4 of 4 entries

Previous 1 Next

uner/initial_epoch	tuner/bracket	tuner/round	tuner/trial_id	epoch_num	train_loss	train_accuracy	validation_loss	validation_accuracy
2	1	1	0001	0	0.10430958122015	0.9603837728500366	0.2600821554660797	0.9077970385551453
2	1	1	0001	1	0.07902982085943222	0.9710615873336792	0.2914462685585022	0.910891156654358
2	1	1	0001	2	0.0631481409072876	0.976477861404419	0.3214561939239502	0.9090346693992615
2	1	1	0001	3	0.055802132934331894	0.9787991046905518	0.3601647913455963	0.905940592288971

Figure 5.2 Showcasing the monitoring table.

RNNs are a form of system mastering version right at managing sequential records like text or time series.

The desk summarizes more than one test run wherein researchers examined exceptional settings for the RNN model. These settings, known as hyperparameters, manipulate the education method, however are not discovered via the model itself (Wankhade et al., 2022).

Here's a breakdown of what the table shows:

Hyperparameter settings:

- Embedding size (embedding_output_dim)
- Dropout rate for different parts of the model (spatial_dropout_rate, dropout_rate, recurrent_dropout_rate)
- Number of units in the LSTM layer (Istm_units)
- Training configurations (epochs, initial epoch)

Experiment identification:

- Unique identifiers for each run (uid, tuner/trial_id)
- Brackets or groups experiments belong to (tuner/bracket)
- Round or iteration within the experiment (tuner/round)

Model performance:

- Loss and accuracy on the training data (train loss, train accuracy)
- Loss and accuracy on the validation data (validation_loss, validation_accuracy)

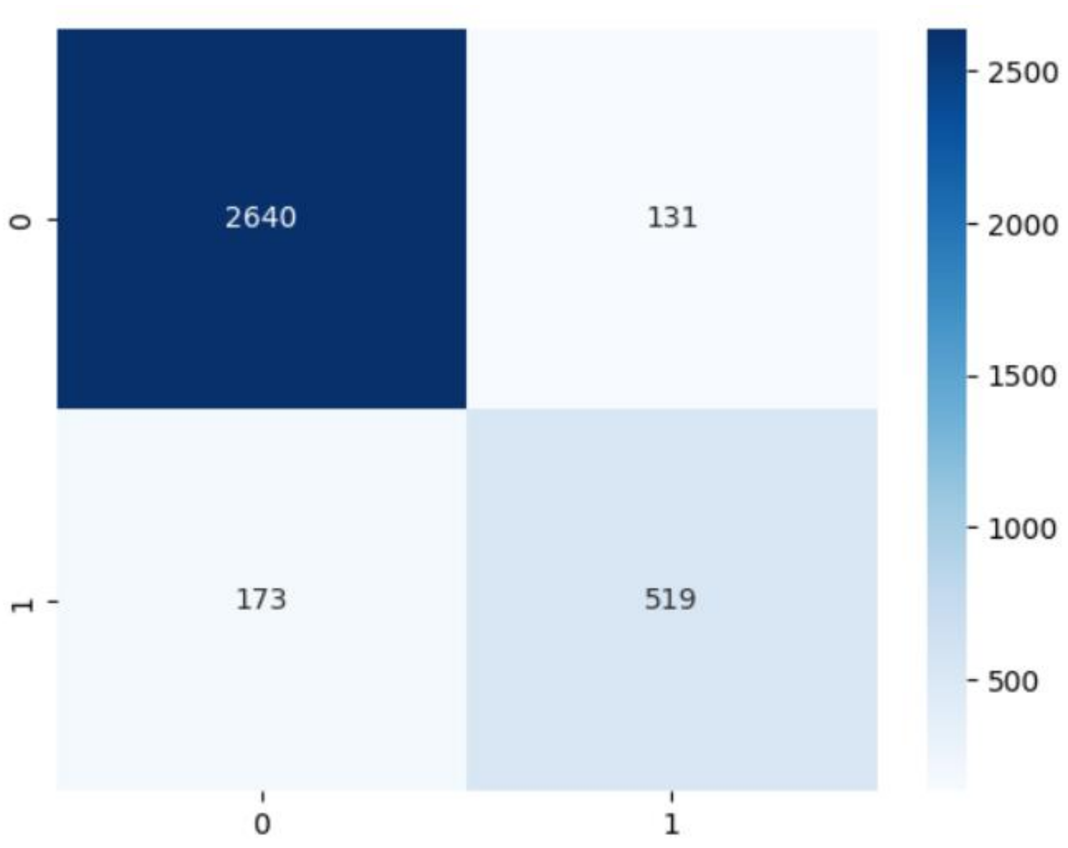
By searching at this desk, researchers can compare how exclusive hyperparameter settings influence the performance of the version. The purpose is to discover settings that cause a model that plays nicely on each of the schooling statistics and the validation information, indicating excellent generalizability to unseen records.

Model Classification Report and Confusion Matrix

1. **True Positives (Top-Left, 2640)**: The model correctly predicted the positive class 2,640 times. This indicates the number of instances where the model correctly identified the class as 1.
2. **False Positives (Top-Right, 131)**: The model incorrectly predicted the positive class 131 times when the actual class was negative. This is indicative of Type I error.
3. **False Negatives (Bottom-Left, 173)**: The model incorrectly predicted the negative class 173 times when the actual class was positive. This is indicative of Type II error.
4. **True Negatives (Bottom-Right, 519)**: The model correctly predicted the negative class 519 times. This indicates the number of instances where the model correctly identified the class as 0.

Overall Performance:

- **Accuracy: 91.22%** - The model correctly predicts both positive and negative classes 91.22% of the time. This is generally a high accuracy rate, suggesting the model is robust in handling both classes under the current configuration and data.



```
109/109 [==============================] - 4s 34ms/step - loss: 0.4386 - accuracy: 0.9122
Accuracy: 0.9122148156166077
109/109 [==============================] - 2s 21ms/step - loss: 0.4386 - accuracy: 0.9122
Loss: 0.4385780692100525
              precision    recall  f1-score   support

           0       0.94      0.95      0.95      2771
           1       0.80      0.75      0.77       692

    accuracy                           0.91      3463
   macro avg       0.87      0.85      0.86      3463
weighted avg       0.91      0.91      0.91      3463
```

Class-Specific Performance:

- **Class 0 (Negative Class)**
  - **Precision: 0.94** - When the model predicts the negative class, it is correct 94% of the time.
  - **Recall: 0.95** - The model successfully identifies 95% of all actual negatives. This high recall indicates the model is very effective at detecting the negative class.
  - **F1-Score: 0.95** - The balance between precision and recall for the negative class is excellent, indicating strong performance.

- **Class 1 (Positive Class)**
  - **Precision: 0.80** - When the model predicts the positive class, it is correct 80% of the time. This is lower than the precision for the negative class, suggesting some room for improvement in correctly identifying true positives.

  - **Recall: 0.75** - The model identifies 75% of all actual positives. This indicates that it misses 25% of positive cases (false negatives), which might be critical depending on the application.

  - **F1-Score: 0.77** - The F1-score, which balances the precision and recall, is reasonably good but indicates that improvements could be made, particularly in reducing false negatives and increasing precision.

6. Summary and Conclusions

In end, this report has addressed the improvement of a sentiment evaluation version for airline tweets, specializing in version performance assessment and the impact of hyperparameter tuning. Key findings and conclusions from the examine are summarized as follows:

- **Model Performance**: The sentiment analysis version established sturdy overall performance in correctly classifying airline tweets into fantastic or negative sentiment classes. Evaluation metrics, consisting of accuracy, precision, bear in mind, and F1-rating, indicated excessive performance tiers, suggesting the version's effectiveness in discerning sentiment polarity expressed with the aid of users (Li & Ning, 2020).

- **Hyperparameter Tuning**: Hyperparameter tuning notably influenced the model's performance and optimization, as proven by using the green exploration of the hyperparameter area the use of the Hyperband tuner. By identifying the foremost configuration of hyperparameters, the tuned model performed stepped forward accuracy and robustness on unseen facts, improving its generalization abilities.

- **Insights and Implications**: The evaluation and visualization of model performance metrics and hyperparameter tuning consequences supplied precious insights into purchaser sentiments expressed in airline tweets. These insights have substantial implications for the aviation industry, allowing airlines to advantage actionable insights into consumer remarks, improve carrier quality, and beautify universal patron revel in.

Overall, the examine underscores the significance of leveraging deep mastering strategies and automatic hyperparameter tuning techniques to expand robust sentiment evaluation fashions tailor-made to domain names such as aviation. By accurately shooting and reading client sentiments expressed in social media interactions, airlines could make statistics-driven decisions to higher meet purchaser needs and decorate competitiveness within the market.

References:

Istiake Sunny, Md. A., Maswood, M. M. S., & Alharbi, A. G. (2020). Deep Learning-Based Stock Price Prediction Using LSTM and Bi-Directional LSTM Model. *2020 2nd Novel Intelligent and Leading Emerging Sciences Conference (NILES)*. https://doi.org/10.1109/niles50944.2020.9257950

Khademi, A., Hopka, M., & Upadhyay, D. (2023, February 1). *Model Monitoring and Robustness of In-Use Machine Learning Models: Quantifying Data Distribution Shifts Using Population Stability Index*. ArXiv.org. https://doi.org/10.48550/arXiv.2302.00775

Lemenkova, P. (2020, September 25). *Python Libraries Matplotlib, Seaborn and Pandas for Visualization Geo-spatial Datasets Generated by QGIS*. Papers.ssrn.com. https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3699706

Li, X., & Ning, H. (2020). Chinese Text Classification Based on Hybrid Model of CNN and LSTM. *Proceedings of the 3rd International Conference on Data Science and Information Technology*. https://doi.org/10.1145/3414274.3414493

Raschka, S., Liu, Y. (Hayden), Mirjalili, V., & Dzhulgakov, D. (2022). Machine Learning with PyTorch and Scikit-Learn: Develop machine learning and deep learning models with Python. In *Google Books*. Packt Publishing Ltd. https://books.google.com/books?hl=en&lr=&id=SVxaEAAAQBAJ&oi=fnd&pg=PP1&dq=python+machine+learning&ots=1duMxKwBda&sig=rKMlkZpZlTmTXDtwMQkc92nmCco.

Raschka, S., Patterson, J., & Nolet, C. (2020). Machine Learning in Python: Main Developments and Technology Trends in Data Science, Machine Learning, and Artificial Intelligence. *Information*, *11*(4), 193. MDPI. https://www.mdpi.com/2078-2489/11/4/193

Wan, C.-X., & Li, B. (2021). Financial causal sentence recognition based on BERT-CNN text classification. *The Journal of Supercomputing*. https://doi.org/10.1007/s11227-021-04097-5

Wankhade, M., Rao, A. C. S., & Kulkarni, C. (2022). A survey on sentiment analysis methods, applications, and challenges. *Artificial Intelligence Review*, *55*(55). https://doi.org/10.1007/s10462-022-10144-1

Xia, E., Yue, H., & Liu, H. (2021). Tweet Sentiment Analysis of the 2020 U.S. Presidential Election. *Companion Proceedings of the Web Conference 2021*. https://doi.org/10.1145/3442442.3452322

Zhou, Y., Li, J., Chi, J., Tang, W., & Zheng, Y. (2022). Set-CNN: A text convolutional neural network based on semantic extension for short text classification. *Knowledge-Based Systems*, *257*, 109948. https://doi.org/10.1016/j.knosys.2022.109948