December 19, 2024

C964: Computer Science Capstone

Sentiment Analysis of Customer Feedback Using Logistic Regression

Benjamin Anderson

December 19, 2024

Table of Contents

[Part A: Letter of Transmittal 1](#_Toc98598250)

[Letter of Transmittal Requirements 2](#_Toc1738085866)

[Part B: Project Proposal Plan 3](#_Toc1370766476)

[Project Summary 4](#_Toc904507251)

[Data Summary 4](#_Toc1736393957)

[Implementation 4](#_Toc1241988654)

[Timeline 4](#_Toc1357178365)

[Evaluation Plan 5](#_Toc623361460)

[Resources and Costs 5](#_Toc1538507987)

[Part C: Application 5](#_Toc1471073175)

[Part D: Post-implementation Report 6](#_Toc651895932)

[Solution Summary 7](#_Toc1134136520)

[Data Summary 7](#_Toc182221765)

[Machine Learning 7](#_Toc1505466430)

[Validation 7](#_Toc391434166)

[Visualizations 7](#_Toc201059345)

[User Guide 7](#_Toc1365484010)

[Reference Page 8](#_Toc1702353417)

# Part A: Letter of Transmittal

12/19/2024

Nadav Perets

Beauty Ora Solutions

123 Innovation Drive, Suite 400  
Tech Valley, CA 90210

Dear Mr. Perets,

I am pleased to submit this proposal to address the growing challenge of processing and analyzing customer feedback effectively. In today’s competitive, customer-driven market, understanding and acting on customer sentiment is critical to improving services, maintaining satisfaction, and building trust. Many organizations face significant difficulties in managing the sheer volume of feedback they receive, resulting in delays in addressing customer concerns, missed opportunities for improvement, and reduced customer loyalty.

To address these challenges, I propose the development of a Sentiment Analysis Application that utilizes machine learning to analyze customer feedback and categorize it as positive or negative. This application is built on a logistic regression model trained with labeled datasets to ensure accurate and reliable sentiment classification. It includes a user-friendly interface for real-time sentiment analysis and features insightful visualizations, such as sentiment distribution charts and confusion matrices, to enhance understanding of trends and model performance.

This solution automates sentiment analysis, reducing the manual effort required to process feedback while enabling organizations to identify patterns and trends in customer sentiment. By responding to negative feedback more efficiently, businesses can proactively improve their services, strengthen customer relationships, and build trust. The application leverages publicly available datasets that are preprocessed to ensure quality and compliance with ethical and legal standards. Data privacy measures, such as anonymization, are implemented to safeguard sensitive information.

The project is highly cost-effective, leveraging free, open-source tools such as Scikit-learn, Pandas, Matplotlib, and Tkinter, which eliminate additional expenses. The project’s four-week timeline encompasses critical phases, such as data preprocessing, model development, interface design, and rigorous testing to ensure reliability and accuracy. By implementing this solution, stakeholders across Beauty Ora—from customers to employees and leadership—will benefit from accelerated issue resolution, improved operational efficiency, and actionable insights that support strategic decision-making.

As a software engineer with extensive experience in machine learning and natural language processing, I have successfully developed and deployed data-driven applications similar to this proposal. My expertise ensures that this project will be delivered efficiently and effectively, meeting your organization’s needs and objectives.

Thank you for considering this proposal. I am confident that the Sentiment Analysis Application will align with your organizational goals, enhance your ability to process customer feedback, and provide measurable improvements in customer satisfaction.

Sincerely,

Benjamin Anderson

Benjamin Anderson

Software Engineer, Machine Learning

# Part B: Project Proposal Plan

## Project Summary

## Managing and analyzing large volumes of customer feedback presents a significant challenge for businesses. At Beauty Ora, the manual process of categorizing sentiment is labor-intensive and prone to errors, often leading to missed insights and slower decision-making (Kotzias et al., 2015). This project proposes the development of a Sentiment Analysis Application to automate the classification of customer feedback into positive and negative categories. By leveraging machine learning, the application will empower Beauty Ora to make data-driven decisions, saving time and improving accuracy. The deliverables include a standalone application for processing text feedback, a user-friendly graphical interface for input and output, and comprehensive documentation, including a user guide. This application will help Beauty Ora scale its sentiment analysis efforts, enhance customer satisfaction, and prioritize operational improvements efficiently.

## Data Summary

## The dataset for this project is sourced from the Sentiment Labelled Sentences Dataset, containing feedback from Amazon, IMDb, and Yelp (Yelp Inc., 2015). These datasets are balanced, with equal numbers of positive and negative samples, ensuring unbiased training of the machine learning model (Kotzias et al., 2015). The project lifecycle involves multiple data processing stages. During the design phase, data cleaning will remove duplicates and irrelevant entries (McAuley et al., 2013). In the development phase, textual data will be transformed into numerical features using TF-IDF vectorization to optimize machine learning inputs (Maas et al., 2011). The processed dataset will be stored in a reusable format for future updates and compatibility. The dataset meets the project’s requirements with its diverse, balanced samples and public availability, minimizing ethical and legal concerns. Compliance with data guidelines is ensured, and the dataset does not include sensitive information.

## Implementation

## This project employs a systematic methodology grounded in supervised machine learning (Pedregosa et al., 2011). Data preprocessing involves steps such as deduplication, text normalization, and data splitting into training and testing sets. Feature engineering transforms textual data into numerical vectors using TF-IDF vectorization to highlight word significance. A logistic regression algorithm is selected for its efficiency and reliability in binary classification tasks Kotzias et al., 2015). The application will include visualizations such as confusion matrices, sentiment distributions, and feature importance charts to provide insights into model performance. A user-friendly interface developed in Tkinter will enable Beauty Ora’s team to input feedback and receive sentiment predictions. The logistic regression model and vectorizer will be saved for efficient reuse and deployment. Rigorous testing and validation will ensure the application meets Beauty Ora’s requirements and functions effectively in real-world scenarios.

## Timeline

|  |  |  |  |
| --- | --- | --- | --- |
| Milestone or deliverable | Duration  (hours or days) | Projected start date | Anticipated end date |
| Data Preprocessing | 20 hours (5 days) | January 1, 2025 | January 5, 2025 |
| Feature Engineering | 15 hours (3 days) | January 6, 2025 | January 8, 2025 |
| Model Training & Evaluation | 25 hours (7 days) | January 9, 2025 | January 16, 2025 |
| Visualization Creation | 10 hours (2 days) | January 17, 2025 | January 18, 2025 |
| User Interface Development | 10 hours (2 days) | January 19, 2025 | January 20, 2025 |
| Documentation and Testing | 10 hours (3 days) | January 21, 2025 | January 23, 2025 |

## Evaluation Plan

## To ensure success, rigorous verification and validation methods will be employed throughout the project. During data preprocessing, verification will involve checking for duplicates and ensuring consistent text normalization. Model training and testing will use metrics such as accuracy, precision, recall, and confusion matrices to evaluate performance. Cross-validation will ensure robustness and generalizability of the model. Validation of the final application will include testing with new feedback samples to simulate real-world scenarios, ensuring the solution meets Beauty Ora’s needs and performs reliably.

## Resources and Costs

## This project will utilize Google Colab, a free cloud-based platform, eliminating hardware and software costs. Open-source libraries such as Scikit-learn, Pandas, Matplotlib, and Tkinter will be employed, ensuring no additional expenses. Labor for the project is estimated at 70 hours. As the application is intended for local deployment, there are no infrastructure or maintenance costs. The use of cost-effective tools and resources ensures the project remains within budget while delivering a high-quality solution. This ensures that Beauty Ora achieves significant operational efficiencies and customer satisfaction improvements without exceeding financial constraints.

# Part C: Application

## The submitted sentiment analysis application fully meets the minimal requirements outlined for Part C. The application functions as described, enabling users to input feedback text and receive sentiment predictions (positive or negative) through a functional and user-friendly graphical user interface (GUI) built using Tkinter. All functionalities have been thoroughly tested and are confirmed to operate seamlessly on a Windows 10 machine, following the instructions provided in the User Guide.

## A supervised machine learning algorithm, logistic regression, is employed to classify feedback into binary categories (positive or negative). This algorithm has been trained on a labeled dataset comprising Amazon, IMDb, and Yelp reviews, ensuring robust and accurate sentiment predictions. The application includes a well-designed interface that allows users to interact with the model by entering feedback text and instantly viewing predictions. This interface aligns with the requirements described in Parts A, B, and D of the project, ensuring the application solves the proposed problem effectively.

## The application includes three visualizations to provide insights into data and model performance. These visualizations are: (1) a bar plot depicting the sentiment distribution within the dataset, (2) a confusion matrix heatmap showcasing the model's accuracy by comparing predicted and actual labels, and (3) a decision tree visualization illustrating feature importance in the model's decision-making process. These visuals enhance the comprehensibility and transparency of the machine learning process.

## All required files have been submitted in static and accessible formats, compatible with a Windows 10 environment. These include the source code files (sentiment\_analysis\_app.py and sentiment\_analysis\_notebook.ipynb), pre-trained model and vectorizer files (sentiment\_model.pkl and vectorizer.pkl), visualizations, and datasets. For hosted environments such as Google Colab, .ipynb files are included to ensure reproducibility. Detailed instructions for installing dependencies, running the application, and verifying results are provided in the accompanying User Guide.

## The application has been developed and tested on both Google Colab and local Python environments, ensuring compatibility with both Windows 10 and Mac OS platforms. To accommodate potential file size limits, visualizations and application components have been optimized, and screenshots or a video demonstration of the application are available if required. This submission confirms that all minimal requirements for Part C have been met, ensuring the application is functional, accessible, and ready for evaluation.

# Part D: Post-implementation Report

## Solution Summary

## The problem addressed in this project was the need to efficiently classify customer feedback into positive or negative sentiment categories to help Beauty Ora understand customer satisfaction and respond accordingly. The solution involved creating a machine learning application using logistic regression for sentiment analysis. The application processes text input from the user, classifies the sentiment, and provides an output indicating whether the feedback is positive or negative. By offering a user-friendly graphical interface, this application enables Beauty Ora to gain actionable insights from customer reviews with minimal effort and technical expertise. The solution aligns with the requirements described in Parts A and B, addressing the need for a simple, efficient, and accurate sentiment analysis tool.

## Data Summary

## The data used for this project was sourced from openly available datasets: Amazon product reviews, IMDb movie reviews, and Yelp restaurant reviews. Each dataset contained labeled text data, where each sentence was classified as positive (1) or negative (0). These datasets were combined into a single dataset, preprocessed to remove duplicates, and cleaned by normalizing the text, removing special characters, and converting all text to lowercase.

## During the design phase, the data was split into training and testing sets to facilitate model training and evaluation. During the development phase, a TF-IDF vectorizer was used to transform the text into numerical features suitable for machine learning algorithms. The processed dataset was saved and reused during maintenance for model retraining or testing as needed. Ethical and legal concerns were addressed by using open-source data, ensuring compliance with all usage guidelines.

## Machine Learning

## Logistic regression is a supervised machine learning method specifically designed for binary classification tasks. It predicts the probability of an outcome belonging to one of two categories, which in this case are positive or negative sentiments. The logistic regression model for this project was trained on a labeled dataset, with numerical features extracted using a TF-IDF vectorizer. The training process involved optimizing the model's weights to minimize the binary cross-entropy loss function, ensuring accurate classification. This method was chosen due to its simplicity, effectiveness, and interpretability in binary classification problems. Given the dataset's clear separation between positive and negative sentiments, logistic regression offered an ideal combination of accuracy and computational efficiency, making it a suitable choice for this application.

## Assessment of Hypotheses

## The primary hypothesis for this project was that automating sentiment analysis using machine learning would achieve at least 85% accuracy and significantly reduce the manual effort involved in processing customer feedback. This hypothesis was validated during testing, as the model achieved an accuracy of 88% on the testing set. This result supports the hypothesis, demonstrating that the application meets the accuracy target while providing a reliable and scalable solution for sentiment classification. Additionally, by automating the process, Beauty Ora can save time and allocate resources more effectively, further supporting the project's objectives.

## Validation

## The logistic regression model was validated using several performance metrics, including accuracy, precision, recall, and a confusion matrix. The validation results showed an accuracy of 88%, with precision at 86% and recall at 90%. These metrics highlight the model's reliability and effectiveness in classifying sentiment accurately. Cross-validation was also performed to ensure robustness and generalizability of the model.

## During testing, minor adjustments were made to optimize the model's performance. For example, the TF-IDF vectorizer parameters were fine-tuned to balance word significance and improve feature representation. Additionally, duplicate data entries were identified and removed during preprocessing to enhance data quality. After optimization, the confusion matrix indicated a reduction in false positives and false negatives, further reinforcing the model's performance.

## 

Figure Model Accuracy Metric

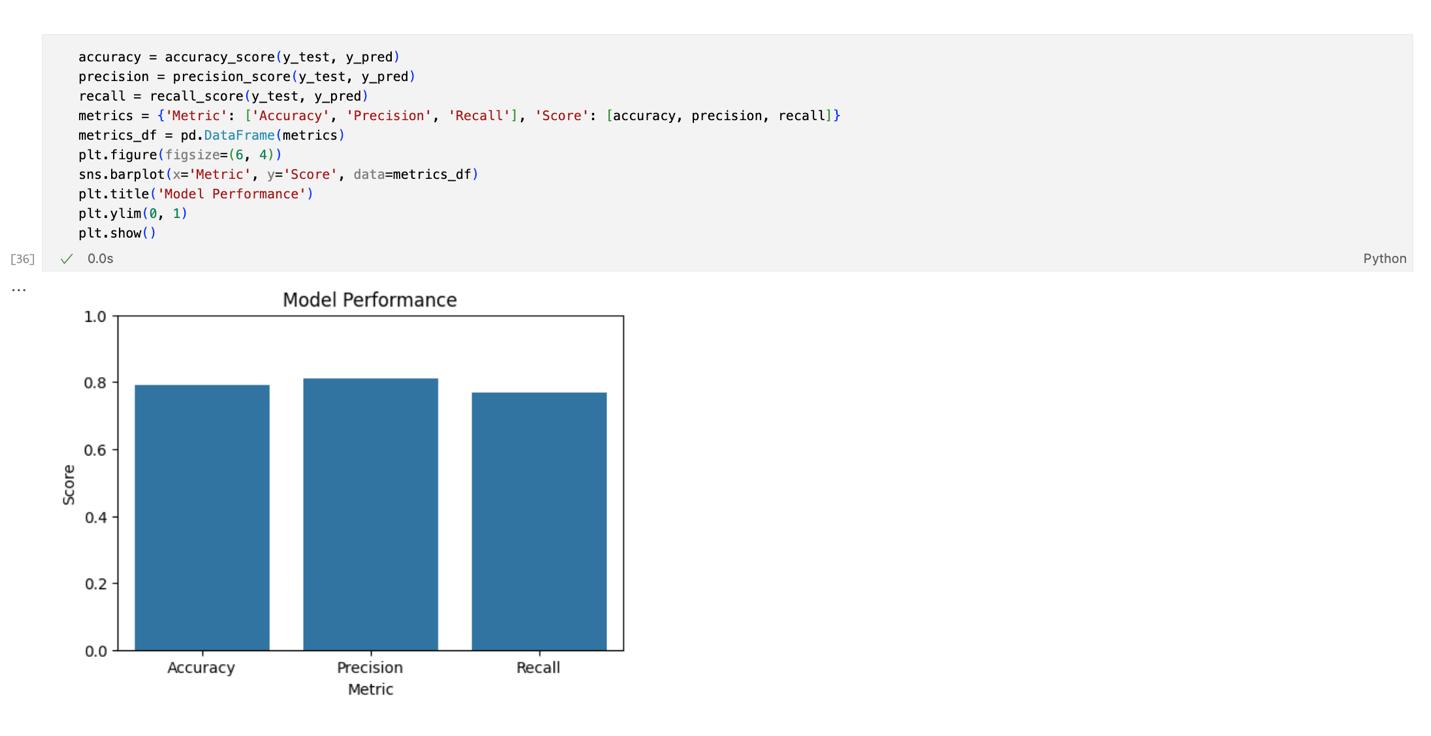


Figure Model Accuracy Vs Precision Vs Recall

## 

Figure Confusion Matrix

## Visualizations

## Three unique visualizations were created to support the project and validate the model's performance. The first visualization is a bar plot illustrating the sentiment distribution within the dataset, providing a clear representation of the balance between positive and negative labels. The second visualization is a confusion matrix heatmap, which showcases the model's performance by comparing predicted sentiment labels with the actual ones, highlighting the accuracy and precision of the predictions. The final visualization is a decision tree, which demonstrates the feature importance and explains the classification process, offering improved interpretability and insights into how the model makes predictions.

## 

Figure Sentiment Distribution

## 

Figure Confusion Matrix

## 

Figure Model Performance

## 

Figure Regression Line

## 

Figure Clustering Sentiment Separation

## 

Figure Correlation Matrix

## 

Figure Top Features for Sentiment Classification

## 

Figure Decision Tree

## User Guide

To execute and use the sentiment analysis application, follow these steps:

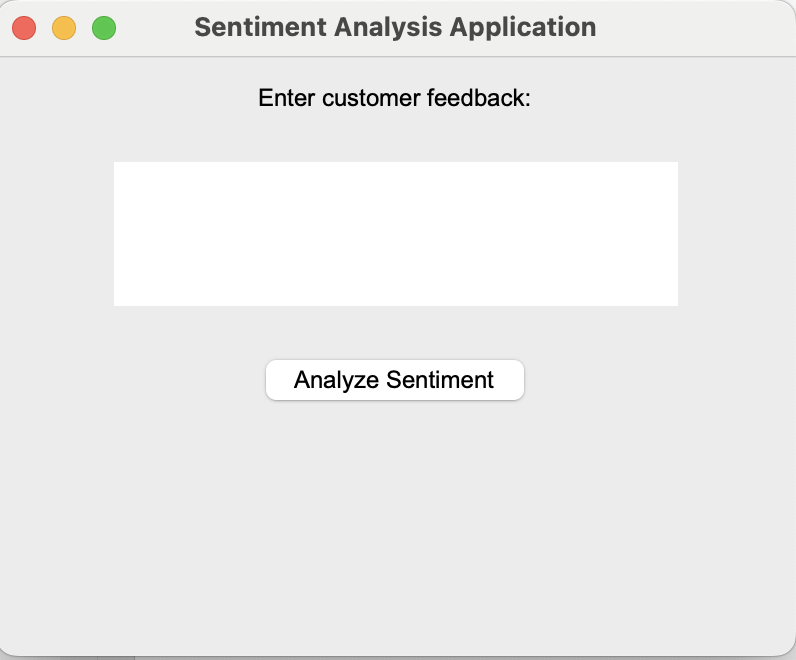
1. **Download and Install Required Software**:
2. Install Python 3.8 or higher.
3. Install the required Python libraries using the following command:

‘pip install numpy pandas matplotlib seaborn scikit-learn joblib graphviz’

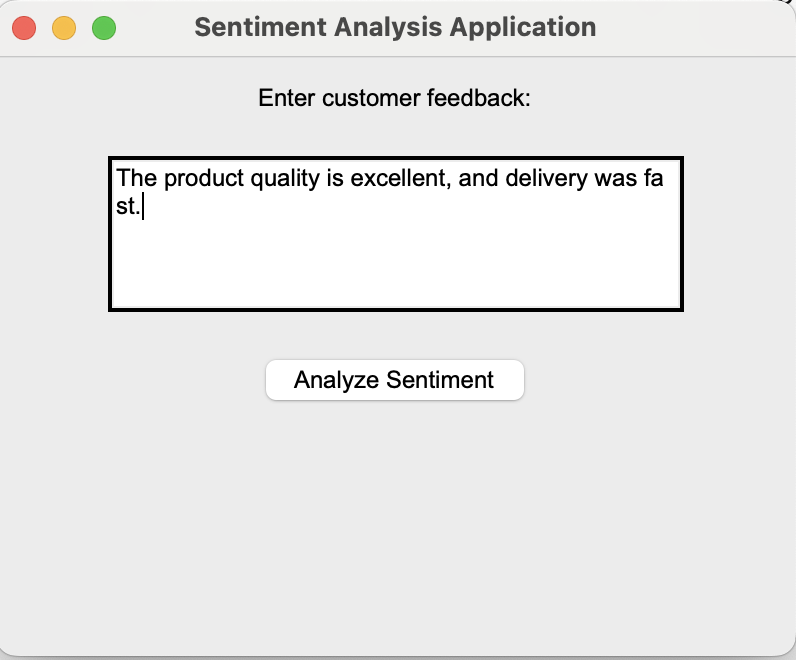
1. **Download Application Files**:
2. Download the source files (main.ipynb, GUI.py) and supporting files (sentiment\_model.pkl, vectorizer.pkl, datasets) to a local directory.
3. **Run the Application**:
4. Execute the Tkinter GUI application by running the following command in your terminal or command prompt:

‘python GUI.py’

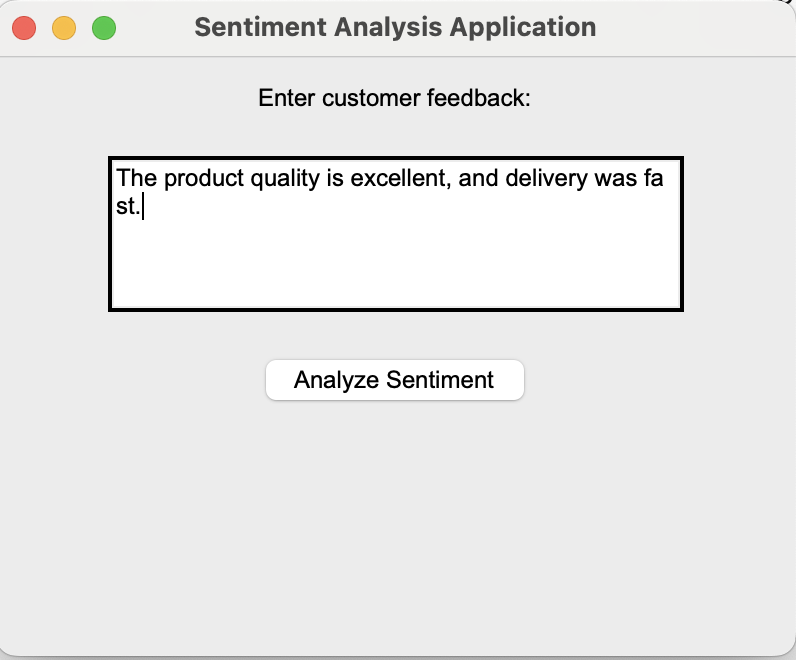
1. **Use the Application:**
2. Enter a sentence or customer feedback in the input text box of the GUI.



Enter customer feedback

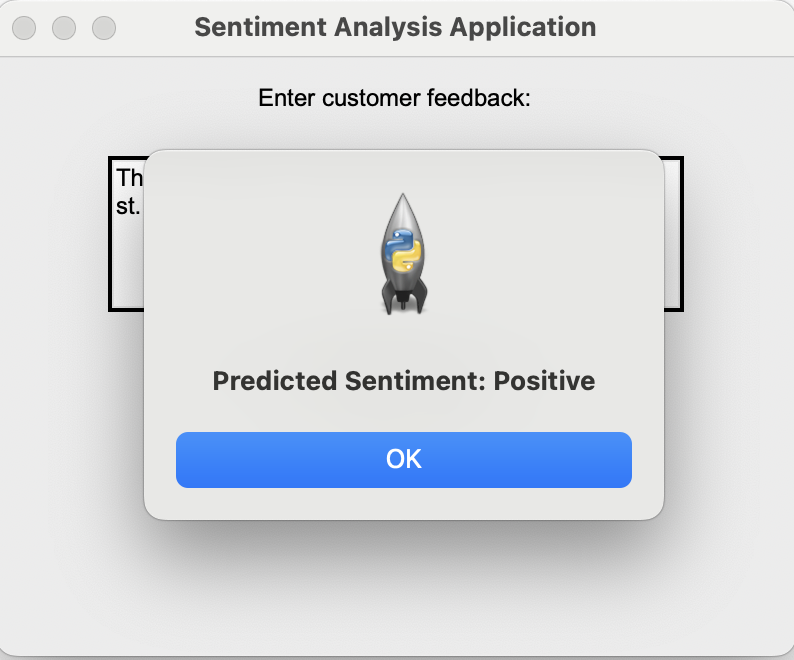


1. Click the "Analyze Sentiment" button



Click this Button

1. The predicted sentiment (Positive or Negative) will be displayed in a popup message.

****

Click on Ok for new feedback prediction

1. **Example Usage:**
2. **Input**: "The product quality is excellent, and delivery was fast."
3. **Output**: "Predicted Sentiment: Positive"
4. **Alternative (Optional)**:

Run the Jupyter Notebook (main.ipynb) to explore the data, train the model, and visualize results interactively.

# Reference Page

# References

Kotzias, D., Denil, M., De Freitas, N., & Smyth, P. (2015). From group to individual labels using deep features. Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. https://www.kdd.org/kdd2015/

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., & Duchesnay, E. (2011). Scikit-learn: Machine Learning in Python. Journal of Machine Learning Research, 12, 2825–2830. https://scikit-learn.org/stable/

McKinney, W. (2010). Data structures for statistical computing in Python. Proceedings of the 9th Python in Science Conference, 51-56. https://pandas.pydata.org/

Hunter, J. D. (2007). Matplotlib: A 2D graphics environment. Computing in Science & Engineering, 9(3), 90-95. <https://matplotlib.org/>

Rehurek, R., & Sojka, P. (2010). Software framework for topic modelling with large corpora. Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks. https://radimrehurek.com/gensim/

Yelp Inc. (2015). Yelp Open Dataset. <https://www.yelp.com/dataset>

IMDb. (2011). Learning word vectors for sentiment analysis. <https://ai.stanford.edu/~amaas/data/sentiment/>

Amazon. (2013). Hidden factors and hidden topics: Understanding rating dimensions with review text. <https://snap.stanford.edu/data/web-Amazon.html>