University of Niagara Falls Canada
Master of Data Analytics
DAMO630 – Advanced Data Analytics
Assignment 2
Due date: Sunday of Week 9, 11:59 PM

Learning outcomes

1. Apply classical machine learning algorithms for sentiment classification tasks.
2. Implement pretrained transformer models to perform sentiment analysis on real-world text.
3. Evaluate sentiment classification models using standard metrics.
4. Build recommendation systems using machine learning models
5. Analyze similarities between items using cosine similarity.

Business Challenge 1: Application of Customer Sentiment Analysis on Yelp Dataset

Yelp is an online platform where people can rate, review, and share experiences about local businesses. Customer reviews contain valuable insights into customer satisfaction, product quality, and service delivery. However, the massive volume makes it impossible for managers to manually read and interpret all reviews. The company is therefore exploring automated sentiment analysis to understand customer opinions at scale and make data-driven decisions that improve service quality and competitiveness.

Consider your self in the analytics team of Yelp and conduct the following tasks

Task I: Understanding Customer Feedback (Exploratory Analysis)
Before building models, the analytics team must understand the structure of customer feedback:

1. Assess whether there is a class imbalance (e.g., more positive reviews than negative).
2. Extract and display sample reviews from both positive and negative categories to illustrate customer voice.
3. Plot the distribution of review lengths to evaluate typical review detail and prepare for model design.

Task II: Building a Baseline Sentiment Model
The company wants an initial benchmark model to classify reviews as positive or negative.

1. Develop a baseline sentiment classifier using TF-IDF + Random Forest.
2. Provide a validation report on a validation partition of the training set showing precision, recall, F1-score, and support to assess strengths and weaknesses of the baseline model.

3. Display a confusion matrix on the validation data to illustrate misclassifications.
4. Use grid search optimization to improve the baseline by tuning:
    - Number of estimators [100, 500]
    - Maximum tree depth [None, 20, 50]
    - Minimum samples split [2, 5]
    - Minimum samples per leaf [2, 4]
    - Maximum features at each node ["sqrt", "log2"]
5. Report the best model parameters and explain why they are important for business decision-making.

Task III: Advanced Sentiment Intelligence with Pretrained Models

The company's leadership is interested in how cutting-edge AI could outperform the baseline.

1. Implement two transformer-based pretrained models (from Hugging Face) for sentiment classification and discuss why you selected each model.
2. Compare their performance against each other and against the optimized Random Forest model.
3. Quantify how much better the transformer models are and explain the business impact (e.g., fewer false negatives could prevent overlooking unhappy customers, better precision could reduce escalation costs).

Deliverable

A Notebook presenting the Python script that develops the followings:

- Data insights from the exploratory analysis.
- A baseline sentiment model and its optimized version via grid search.
- Performance comparison with pretrained AI models, with clear interpretation of the business value.

Business Challenge 2: Books recommendation system development

Digital book platform (similar to Goodreads or Amazon Kindle) always faces the business challenge of how identify similar books from as well as how books can be recommended to readers that they are most likely to enjoy, based on their past ratings and the behavior of other users.

The Goodbooks-10k dataset available in this link, is a popular dataset used for benchmarking recommender systems. The dataset contains customer rating on thousands of books. The file dataset can be briefly described as follows:

- user_id: An anonymized identifier for each reader.
- book_id: An identifier for each book in the collection (covering 10,000 unique books).
- rating: A rating given by the user to the book, ranging from 1 (lowest) to 5 (highest).

- Over 6 million ratings provided by more than 50,000 users.
- Most users have rated only a handful of books, and most books are rated by only a subset of users.

Using this dataset conduct recommendation system analysis by following the tasks given below

Task I: Exploratory Data Analysis (EDA)
Perform exploratory data analysis on the provided dataset.
- Summarize the dataset: number of rows, columns, and types of variables.
- Show the distribution of labels or ratings.
- Provide at least two visualizations (e.g., histogram, bar chart, word cloud).

Task II: Apply the collaborative filtering model, Alternating Least Squares (ALS).
The model is available in the implicit python library. You may use "pip install implicit" to install the library.

a) Explain the ALS model in your own words.

- What problem does it solve?
- How does it use matrix factorization to recommend items?
- What are its main hyperparameters?

b) Train the ALS model on a training set only.
- Convert the dataset into a user–item interaction matrix.
- Fit the ALS model using the implicit library.
- Report factors, regularization, iterations, and any other key parameters.

Deliverables

• Submit a Jupyter Notebook with code, outputs, and explanations.
• Provide clear interpretations and explanations of the developed code.

DAMO630- Rubrics of Assignment 2

Criteria	Excellent (85–100%)	Very Good (70–84%)	Good (60–69%)	Fail (<60%)
Exploratory Data Analysis (EDA)	Thorough analysis: class balance clearly reported, representative examples shown for each class, and insightful visualization.	Covers all tasks correctly, but analysis lacks deeper insights or clear link to model implications.	Completes most tasks but superficial analysis (e.g., plots without interpretation).	No meaningful EDA provided.
Baseline Sentiment Model (Random Forest + TF-IDF)	Baseline built correctly with full validation report & confusion matrix; grid search performed systematically; best parameters reported with clear justification in a business context.	Baseline and grid search done; validation/confusion matrix shown, but explanation of parameter importance is brief or technical.	Baseline runs and grid search attempted, but incomplete reporting (e.g., missing confusion matrix or best params).	No baseline model developed.
Advanced Models (Transformers)	Two pretrained Hugging Face models implemented correctly, results compared with baseline, and clear business impact (false negatives, precision gains, etc.) explained.	Two models compared with baseline; performance improvement shown but little business-oriented interpretation.	At least one transformer model tested; limited comparison with baseline.	No transformer models used.
ALS Explanation	Excellent explanation in own words: problem addressed, matrix factorization concept, hyperparameters (factors, reg., iterations, alpha, etc.). Links to recommender system context.	Good explanation but misses one of the key aspects (e.g., hyperparameters).	Partial explanation. Uses textbook-like definition with little context.	Little to no explanation or copy–paste with no understanding.
ALS Implementation (Training)	Correctly converts dataset to user–item CSR matrix. Fits ALS with implicit library. Reports parameters (factors, regularization, iterations, etc.). Code well-documented.	Correct implementation but missing one detail (e.g., partial parameter reporting).	Implementation works but incomplete or poorly explained.	No working code, or major errors in implementation.

Business Insight & Reporting	Clear, professional notebook/report: interprets results in business terms (service quality, customer satisfaction impact), structured well, and reproducible.	Good interpretation and organization; minor gaps in business framing or clarity.	Report explains technical results but weak business linkage.	No coherent reporting or business interpretation.