# Research Proposal: Detecting Manipulated Online Reviews Using Behavioural and Textual Features with Classical Machine Learning

**Student Name:** [Insert Name]

**Student ID:** [Insert ID]

**Module:** 771765 Research and Application in AI and Data Science

**Date:** [Insert Date]

## (a) Aims and Objectives

**Aim**

The primary aim of this research is to design, implement, and rigorously evaluate a hybrid machine learning–based detection system capable of identifying manipulated or "fake" online reviews. Unlike traditional approaches that rely solely on linguistic analysis, this study seeks to bridge the gap between Natural Language Processing (NLP) and Behavioural Forensics. The proposed system will integrate semantic analysis of review content with novel metadata-driven behavioural features to detect deceptive patterns that text-only models frequently miss. Furthermore, this research prioritises *explainability* by utilising interpretable classical machine learning algorithms, addressing the critical industry need for transparent decision-making in fraud detection systems, rather than relying on opaque "black-box" deep learning architectures.

**Objectives**

1. **To systematically analyse patterns of review manipulation** within the Yelp Reviews Dataset by distinguishing between "singleton" spammers (individuals acting alone) and "group" spammers (coordinated campaigns), examining how their linguistic footprints and temporal activities differ from genuine users.
2. **To engineer and validate a multidimensional feature set** that goes beyond standard text vectors. This involves constructing complex behavioural metrics—specifically *Rating Deviation* (calculating the statistical distance from the consensus), *Temporal Burstiness* (identifying rapid-fire review clusters), and *Extreme Rating Entropy*—to serve as high-signal predictors of deception.
3. **To develop and benchmark classical supervised learning classifiers**—specifically Logistic Regression, Multinomial Naive Bayes, and Random Forest. The objective is to determine which algorithm offers the optimal trade-off between *Precision* (minimizing false accusations against legitimate users) and *Recall* (maximizing the detection of actual fraud) in a highly imbalanced dataset.
4. **To quantify feature importance and model explainability** using SHAP (SHapley Additive exPlanations) values. This objective aims to move beyond binary classification to provide actionable insights, specifically answering *which* behavioural attributes (e.g., timing vs. sentiment) are the strongest indicators of manipulation.

## (b) Background and Literature Review

**The Context of Online Review Manipulation**

In the contemporary digital economy, online consumer reviews have evolved into a primary currency of trust. Platforms such as Yelp, Amazon, and TripAdvisor leverage user-generated content to facilitate decision-making; however, this influence has directly monetized reputation. This economic incentive has given rise to "Opinion Spam"—deceptive reviews written by individuals who have not experienced the service, posted with the intent to artificially inflate a business’s reputation (hype spam) or damage a competitor’s standing (defaming spam). The economic stakes are substantial; a Harvard Business School study found that a one-star increase in a Yelp rating can lead to a 5-9% increase in revenue for independent businesses (Luca, 2016). Consequently, the detection of such manipulation has become an adversarial "arms race" between fraudsters employing increasingly sophisticated techniques and researchers developing detection algorithms.

**Limitations of Text-Centric Approaches**

Historically, the academic literature on fake review detection has been dominated by content-based (text-centric) approaches. Early research, such as that by Jindal and Liu (2008), focused heavily on sentiment analysis and n-gram modelling, operating under the assumption that deceptive reviews contained specific lexical markers—such as excessive use of superlatives ("amazing," "best ever") or a distinct lack of spatial detail. While these methods were effective against early, automated bot-generated spam, they face significant diminishing returns against human-generated fake reviews.

A pivotal study by Ott et al. (2011) demonstrated the severity of this limitation. They found that human judges performed barely better than chance (approx. 57% accuracy) at differentiating between genuine reviews and deceptive ones generated by Mechanical Turk workers. This suggests that humans are adept at mimicking the linguistic style of genuine reviews. Furthermore, the recent proliferation of Large Language Models (LLMs) allows malicious actors to generate grammatically perfect, contextually relevant, and unique reviews at scale, rendering linguistic cues alone increasingly unreliable as a sole method of detection. relying solely on *what* is written is no longer sufficient; we must analyse *how* the user behaves.

**The Paradigm Shift to Behavioural Feature Engineering**

To overcome the limitations of text analysis, recent scholarship has pivoted toward behavioural feature engineering. This paradigm posits that while a fraudster can easily fabricate the *content* of a review, it is significantly harder and more resource-intensive to fake the *metadata* and *long-term behavioural patterns* associated with genuine user activity.

Mukherjee et al. (2013) were among the first to systematically categorise these behavioural footprints. They identified that members of spam rings often post reviews in "bursts" (rapid succession) to meet quotas and exhibit "rating deviation"—consistently giving 5-star ratings to businesses that average 2 stars, or vice versa. For instance, a genuine user typically reviews a diverse range of business categories over a long period with a Gaussian distribution of ratings. In contrast, a manipulated account might be created solely to boost a specific set of businesses within a short timeframe (the "lock-step" behaviour). Rayana and Akoglu (2015) expanded on this by utilising graph-based methods to map the network of reviewers, finding that these behavioural anomalies are often far more predictive of fraud than semantic content. This research project builds directly upon these findings by proposing a hybrid model—one that marries the semantic understanding of classical NLP with the statistical rigour of behavioural analysis.

**Justification for Classical Machine Learning**

While Deep Learning (DL) models like BERT and LSTM currently dominate general NLP tasks, this project deliberately selects classical machine learning algorithms (Logistic Regression, Naive Bayes, Random Forest). The rationale is twofold. First, DL models often require massive computational resources and large labelled datasets, which can be prohibitive for initial deployment. Second, and more importantly, DL models function as "black boxes." In the context of fraud detection and compliance, *explainability* is paramount (Aggarwal, 2018). If a system flags a user as a spammer, platform moderators need to understand *why*—whether it was due to the user's account age, the review frequency, or the text itself. Classical models allow for the direct extraction of feature importance, enabling the research to answer the "why" question effectively and providing a transparent audit trail.

## (c) Research Description and Anticipated Outcomes

**1. Dataset Selection and Preprocessing**

The research will utilise the **Yelp Reviews Dataset** sourced from Kaggle. This dataset is an industry standard for this domain because it contains the requisite "ground truth" (filtered reviews flagged by Yelp’s proprietary algorithm) and, crucially, rich metadata. The dataset includes fields for Review Text, Star Rating, Review Date, User ID, and Business ID.

* **Data Cleaning:** The raw data will undergo rigorous cleaning to ensure model validity. This includes removing duplicate entries and, critically, filtering out users with fewer than five reviews (the "cold start" constraint). Behavioural features cannot be reliably calculated for users with no historical footprint; therefore, this study focuses on detecting *repeat offenders* rather than one-off bots.
* **Text Preprocessing:** The textual data will be processed using a standard NLP pipeline via the NLTK or SpaCy libraries:
  + *Tokenisation & Normalization:* Converting text to lowercase and removing punctuation.
  + *Stop Word Removal:* Eliminating high-frequency, low-meaning words (e.g., "the", "is") to reduce noise.
  + *Lemmatization:* Reducing words to their base roots (e.g., "running" to "run") to consolidate feature space.
  + *Vectorisation:* The processed text will be transformed using TF-IDF (Term Frequency-Inverse Document Frequency). This method is chosen over simple Bag-of-Words because it downweights common terms and highlights unique words that may define specific review signatures.

**2. Feature Engineering Strategy**

This project’s core novelty lies in the engineering of a multidimensional feature space that captures the "fingerprints" of deception.

* **A. Textual Features:**
  + *Sentiment Polarity:* Using a lexicon-based approach (VADER or TextBlob) to score the sentiment of reviews on a scale of -1.0 to +1.0. The hypothesis is that fake reviews often exhibit "extreme" sentiment (very high or very low) with less nuance than genuine reviews.
  + *Review Length:* Calculated as the word count. Deceptive reviews are hypothesized to be either suspiciously short (low effort) or unnecessarily verbose (trying too hard to sound convincing).
* **B. Behavioural Features (The Core Contribution):**
  + *Rating Deviation:* This will be calculated mathematically as , where  is user 's rating for item , and  is the average rating of business . A high deviation suggests the reviewer is attempting to skew the average against the consensus.
  + *Temporal Burstiness:* This feature measures the variance in time between a user’s reviews. Spammers often log in, write multiple reviews quickly to fulfill a contract, and then abandon the account. This will be quantified by calculating the inverse standard deviation of the time gaps between a user's consecutive posts.
  + *Extreme Rating Ratio:* The proportion of a user's total reviews that are either 1-star or 5-stars. Genuine users tend to have a wider distribution of ratings; spammers often have a bimodal distribution focused on the extremes.
  + *Maximum Number of Reviews per Day:* A simple count feature to flag superhuman typing speeds or bot-like activity.

**3. Machine Learning Modelling**

Three distinct supervised learning algorithms will be trained and compared to establish a performance baseline and an optimal classifier:

1. **Logistic Regression:** This will serve as the linear baseline. It is computationally efficient and provides a straightforward probability score for the likelihood of a review being fake (Log-Odds).
2. **Multinomial Naive Bayes:** Selected for its probabilistic approach and historical success in spam filtering and text classification tasks where feature independence is assumed. It handles high-dimensional TF-IDF vectors exceptionally well.
3. **Random Forest:** This ensemble method is critical for this study. Unlike linear models, Random Forests can capture non-linear interactions between features (e.g., a review is only suspicious if it is *both* short *and* has high rating deviation). It constructs multiple decision trees during training and outputs the class that is the mode of the classes of the individual trees, reducing the risk of overfitting.

**4. Evaluation and Explainability**

The models will be evaluated using a stratified hold-out validation set (80% training, 20% testing).

* **Metrics:** Given the class imbalance (genuine reviews far outnumber fake ones), **Accuracy** is a misleading metric. Therefore, the project will prioritise **Precision** (to avoid falsely flagging real customers) and **Recall** (to catch as much fraud as possible), combined into the **F1-Score**.
* **Explainability:** To address the "black box" issue, the project will utilize **SHAP (SHapley Additive exPlanations)** values. SHAP provides a unified measure of feature importance, allowing us to quantify the marginal contribution of each feature. For example, we can determine if "Rating Deviation" contributes +15% to the probability of a specific review being classified as fake, providing a granular level of insight that global feature importance plots cannot.

**Anticipated Outcomes**

The successful completion of this project will yield the following outcomes:

1. **A Validated Detection Pipeline:** A complete, end-to-end Python-based system capable of ingesting raw review data, calculating complex behavioural features, and outputting classification labels with associated confidence scores.
2. **Feature Importance Hierarchy:** Empirical evidence ranking the effectiveness of behavioural vs. textual features. It is anticipated that behavioural features (specifically Rating Deviation and Burstiness) will hold higher predictive power than TF-IDF text features, confirming the hypothesis that *metadata* is more reliable than *content*.
3. **Comparative Benchmark:** A documented comparison demonstrating that ensemble methods (Random Forest) outperform linear baselines in this domain due to their ability to model complex decision boundaries.
4. **Actionable Insights for Moderators:** The study will derive interpretable rules (e.g., "Users who deviate from the average rating by >2 stars and review >3 times a week are 80% likely to be deceptive"), which have practical, immediate applications for platform moderators looking to automate fraud detection.

## (d) Project Timeline (Gantt Chart)

The following timeline outlines the structured progression of the research over the trimester. It accounts for iterative phases of development and includes buffer time for risk mitigation.

| **Task / Phase** | **Weeks** | **Details & Milestones** |
| --- | --- | --- |
| **Literature Review & Refinement** | Weeks 1–2 | Update literature review with 2024/25 papers; refine research questions based on latest trends in fraud detection (e.g., LLM-generated spam). |
| **Dataset Acquisition & Exploration** | Week 3 | Download Yelp dataset; perform Exploratory Data Analysis (EDA) to understand class distribution, visualize "cold start" users, and assess data quality. |
| **Data Preprocessing** | Weeks 3–4 | Cleaning data; handling missing values; implementing text normalization pipeline (lemmatization, stop-word removal). |
| **Feature Engineering (Core)** | Weeks 4–6 | **Critical Phase:** Coding the logic for Behavioural Features (Burstiness, Deviation, Entropy) and Textual Features (TF-IDF). |
| **Baseline Model Development** | Weeks 6–7 | Training and testing Logistic Regression and Naive Bayes models to establish a performance baseline. |
| **Advanced Model Development** | Weeks 7–8 | Implementation of Random Forest; hyperparameter tuning (GridSearchCV) to optimize model performance (e.g., number of trees, max depth). |
| **Evaluation & Analysis** | Weeks 8–9 | Generating confusion matrices; calculating F1/Recall; performing error analysis on misclassified reviews to understand model weaknesses. |
| **Explainability (SHAP/Feature Imp.)** | Weeks 9–10 | Implementing SHAP library to visualise feature contributions; interpreting the "black box" to generate feature importance plots. |
| **Results Visualisation** | Week 11 | Creating final graphs, charts, and tables for the dissertation; synthesizing statistical findings. |
| **Dissertation Writing (Drafting)** | Weeks 11–13 | Writing the Introduction, Methodology, and Results chapters; refining the narrative and connecting results to literature. |
| **Final Review & Submission** | Week 14 | Final proofreading; formatting checks against university guidelines; submission to Canvas/Turnitin. |

**Risk Mitigation Strategy:**

* *Data Imbalance:* The Yelp dataset is likely to be heavily skewed towards genuine reviews. To mitigate this, **SMOTE (Synthetic Minority Over-sampling Technique)** will be applied during Week 4 to balance the training classes.
* *Computational Complexity:* If feature engineering (specifically the user-grouping operations) proves computationally expensive, a randomized subset of the data will be used for prototyping before scaling up to the full dataset.

## References

Aggarwal, C.C. (2018). *Machine Learning for Text*. Cham: Springer.

Fang, Y., Zhang, H., Wilcox, G. & Harper, F.M. (2016). Detecting fake reviews using sentiment analysis. *Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics*.

Jindal, N. & Liu, B. (2008). Opinion spam and analysis. *Proceedings of the 2008 International Conference on Web Search and Data Mining (WSDM)*, pp. 219–230.

Li, F., Huang, M., Yang, Y. & Zhu, X. (2011). Learning to identify review spam. *Proceedings of the 22nd International Joint Conference on Artificial Intelligence (IJCAI)*, pp. 2488–2493.

Liu, B. (2020). *Sentiment Analysis: Mining Opinions, Sentiments, and Emotions*. Cambridge: Cambridge University Press.

Luca, M. (2016). Reviews, reputation, and revenue: The case of Yelp.com. *Harvard Business School Working Paper*, 12-016.

Mukherjee, A., Liu, B. & Glance, N. (2013). Spotting fake reviewer groups in consumer reviews. *Proceedings of the 22nd International World Wide Web Conference (WWW)*, pp. 191–200.

Ott, M., Choi, Y., Cardie, C. & Hancock, J.T. (2011). Finding deceptive opinion spam by any stretch of the imagination. *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics (ACL)*, pp. 309–319.

Rayana, S. & Akoglu, L. (2015). Collective opinion spam detection: Bridging review networks and metadata. *Proceedings of the 21st ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*, pp. 985–994.