Team 076: Detection and Visualization of Fraudulent Reviews on Yelp

Jennie Becker, Jason Burkholder, Maithily Erande, Matthew Pless, Christopher Sawtelle, Daniel Snider, Lu Yu

1 Introduction

With the development of the global e-commerce market, shopping consumption is no longer limited by space or time [1]. Online consumer review sites and platforms significantly impact business patterns of merchants, distributors and consumers. It has brought convenience, while increasing customer reliance on online reviews for decision making [2]. It has been reported that 87% of consumers rely on positive reviews to make purchasing decisions. Additionally, around 80% of consumers are likely to change their initial purchasing decisions when confronted with a substantial number of negative reviews on commodities [3]. Unfortunately, because online reviews can influence sales and customer purchasing behavior, it also encourages spammers to produce fake reviews to increase economic benefit or discredit reputation [4]. Fake reviews are prevalent and can harm brand reputation and the overall online shopping environment. Therefore, detection of fake reviews has become a valuable research. This project focuses on the detection and visualization of fraudulent reviews on Yelp. The primary motivation behind conducting this project is to improve the online shopping experience for consumers, maintain the reliability and authenticity of the e-market, and better communicate impacts of fraudulent reviews to consumers and businesses.

2 Problem Definition

A summary of the consequences of fake online reviews emphasized the harm on the quality and credibility of online information [5]. Fake reviews are immediately effective at improving sales, but are also detrimental to both businesses and consumers in the long term [6]. Moreover, even a small portion of fake reviews are sufficient to impact the vulnerability of a business [7]. Businesses with high competition and poor reputations are more likely to engage in review fraud campaigns to boost reputation. Small businesses in particular are particularly prone to commit review fraud [8]. All of these reports emphasized the need for analyzing, identifying, and classifying fake reviews. However, the previous literature may fall short in identifying and communicating which aspects the various models have learned and how this impacts businesses and consumers alike. We intend to address this shortcoming by providing insight into the features that lead to a classification of fraudulent reviews. Doing so is not without risk. Displaying a visual assessment of review classification can be exploited by fraudsters to develop techniques to better avoid detection, and we will attempt to balance this risk with usefulness. Our tool might be of interest to the broader community of consumers and business owners alike. Upon successful completion, we hope to improve the authenticity of the e-market; informing consumers on businesses to avoid while giving business owners incentive for honest reviews. By exposing important features and impacts of fraudulent reviews in an easily digestible way, we can promote confidence in e-commerce market customers.

3 Literature Survey

Kumar et al explored the impact of feature engineering, data pre-processing, and underlying distributional characteristics on detecting fraudulent behavior. Addressing these factors lead to an increase in performance across all tested algorithms [9]. Existing research shows that the main features used for fake review detection are behavioral and textual [10]. These include subjectiveness, informativeness, readability and proportion of spelling errors [11]. Business and spatial features have also proven significant in predicting business and reviewer fraud participation [8, 12, 13]. Additionally graph features have been shown to support prediction of linkage weights in Yelp review data [14]. This may help identify unlikely review activity by incorporating user relationship information. Though little comparison has been done on the effects of reviewer-side and non-reviewer-side features on fake review detection, feature combination from various sources may provide a more complete view of trends present in Yelp reviews. While there exist few instances of effective consumer-targeted visualizations in this field, one relevant study investigates the usefulness of interactive visualization techniques for detection; such as encoding and filtering in detecting fraudulent financial and accounting activity [15]. They acknowledge that "research into the

efficacy of interactive data visualizations for fraud detection is extremely limited to date."

Many types of models including traditional ML models, such as Support Vector Machine(SVM), Naïve Bayes (NB), K-nearest neighbors (KNN), Logistic regression (LR), and Random forest (RF), Neural Network models(CNNs, GAN, RNN, BERT, BiLSTM), and ensemble methods are used today for identifying fake reviews [10, 16]. Traditional machine learning models, such as SVM, Random Forest and KNN have found some success, with SVM generally performing the best among traditional models [12, 17–19]. One paper explored combinations of these types of approaches targeting businesses likely to be disproportionately affected by review fraud [12]. Several different deep learning models have also been attempted, including variations on BERT, BiLSTM, and CNNs, which show dramatic performance improvements over traditional models [10, 20–23]. Another paper utilized change point detection to compare trends in business ratings over time between ratings and reviews providers to identify reviews as "suspicious" [24]. All of the models are significantly more effective at identifying fake reviews compared to human attempts, which peak at around 65% effectiveness and around 81% accuracy for SVM, showing that there are significant benefits to applying modeling to the problem [17, 20]. Harris explored a state-of-the-art ensemble method, reporting that the combination of K-L divergence technique to SVM or LSTM could successfully outperform other techniques on the same or similar data sets [16, 25]. Despite the limitation of only focusing on linguistic features, these papers provide descriptive applied methodology which we can build on. Accuracy, precision, recall, and F-score can be used to monitor and compare the performance across different models[10, 26].

4 Proposed Method

4.1 Intuition

The major reasons we think our method is better than the state of the art are:

1. We have a set of features that we have not seen in previous literature, including the number of reviews by a user in 24 hours, count of punctuation, average sentiment by both user and business, and content similarity to the user's other reviews.

2. Previous research shows several examples of different types of behavioral, textual and word embedding features, however there has been little research on combining these together. In this project, the performance of models combining these sets of features were tested and evaluated.

3. Our research uncovered little about visualizations for fraudulent reviews. While others focus on model construction, application, and performance, we take this further with an interactive spatial visualization. For the end user, this links the technical model output to digestible, real-world impact.

4.2 Detailed Approaches

4.2.1 ***Preprocessing*** Text preprocessing is important for fake review detection as it helps clean the data, removes noise and improves the overall accuracy of the classification.

1. Cleaning: Punctuation and numbers were removed, words were converted to lower case and contractions were replaced with original forms
2. Tokenization: Text strings were split into a list of individual words
3. Stop-word removal: Common words, such as "the", "and", and "a" were removed
4. Lemmatization: Each word is converted to its root. E.g. "apples" becomes "apple", "disappeared" becomes "disappear".

4.2.2 ***Feature engineering*** Typically, creating attributes for reviews involves utilizing text and natural language processing techniques. However, it may be necessary to also involve features associated with the reviewers' behavior for fake review detection. Textual features were derived from the review text, by analyzing semantic, lexicon and meta-data contents [10]. A series of textual features across 6 categories were first extracted from review comments from the YelpZip data set. The selected **textual features** include: **1. Quantity:** number of words, number of verbs, **2. Complexity:** average word length, number of clauses, number of positive words, number of negative words, sentiment score, **3. Emotiveness Ratio:** ratio of emotiveness, **4. Diversity:** lexical diversity, **5. Informality:** typo ratio, **6. Non-Immediacy:** personal pronoun counts, **7. Content Similarity:** average review cosine similarity to the user's other reviews.

A user's behavioral features have been shown to be highly predictive of posting fraudulent reviews [9]. Selected **behavioral features** include: **Rating ratios:**1. Positive review (4-5 star) ratio, Negative review (1-2 star) ratio, and **2. Review upload count and timing:** Number of reviews user posted in 24 hours previous to review, Maximum number of reviews posted by user in a 24 hour period, Total number of reviews posted by user

TF-IDF is a simple but powerful technique that provides insight into which words are most relevant to a particular document and provides critical information to the model when classifying the reviews. This is a high-dimensional feature space, with a separate feature for each unique word in the entire review corpus.

4.2.3 ***Models*** We implemented and tested 5 classifiers using supervised learning algorithms that have been reported for fake text review detection. The performance of all the five models were evaluated, compared and discussed.

1. **Support Vector Machine (SVM)**. SVM is reported to have the best performance on fake review detection among the traditional classification models[12, 17–19], so it's chosen as a baseline model. For text classification, a linear kernel SVM is commonly recommended [27] for its advantage of processing large datasets.

2. **Naïve Bayes (NB)**. NB is a statistical classifier that predicts the probability of a given sample belonging to a class based on Bayes' theorem [28]. It assumes all the features are conditionally independent and assigns a posterior class probability to a sample [29].

3. **K-nearest neighbors (KNN)**.KNN is one of the most powerful and simple classification algorithm for fake review detection[10, 30]. It works to classify new data based on its proximity to training data. The execution time of KNN will be significantly increased as the number of samples increase [31].

4. **Logistic regression (LR)**. LR is another supervised algorithm that's often used to classify data. It depends on the identification of a hyperplane that classifies the reviews as spam or not [30].

5. **Random forest (RF)**. By combining bagging and feature randomness, the random forest algorithm extends the bagging method to create a combination of tree classifiers, known as a forest [32]. This ensemble model is relatively robust to outliers and noise, faster than bagging or boosting [33] and used in fake review detection [10, 34].

4.2.4 ***User Interface*** Our model's results are communicated through a d3.js-built choropleth map featuring interactive drill-down capabilities to the major metro regions represented in the dataset. On initial load, the user is presented with a map of the Indianapolis metro region, depicting by color the percentage of fake reviews identified for all restaurants aggregated at the zip code level.

As shown in figure 1, our visualization consists of four key components:

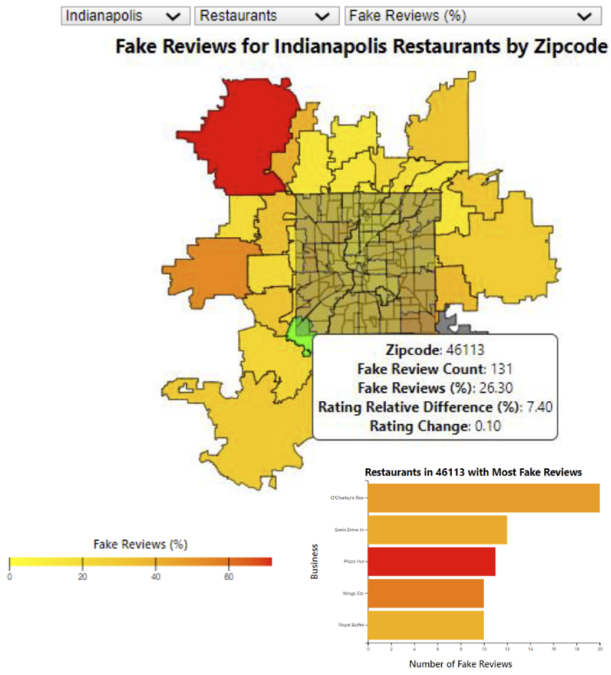

Figure 1: Interactive User Interface: Default View

1. **Drop down menus** which allow the user to navigate among metro regions, business categories, and metrics. Each time a different value is selected, the map updates to reflect it.

2. A **choropleth map** which depicts the selected metric for the selected business category in the selected urban and suburban metro region. The light grey overlay on the map shows the city boundaries with neighborhood dividers. The zip codes outside of the overlay correspond to the suburban metro area.

3. A conditional **bar chart** which displays when a user hovers over the map. It shows five businesses for

the hovered zip code with the highest number of fake reviews as identified by our model.

4. An accompanying **legend** for the chosen metric which indicates the values represented by each color for the chloropleth map and bar chart.

5. A **tool tip** which displays when a user hovers the map. It reveals the zip code, fake review count, and metrics for the hovered zip code.

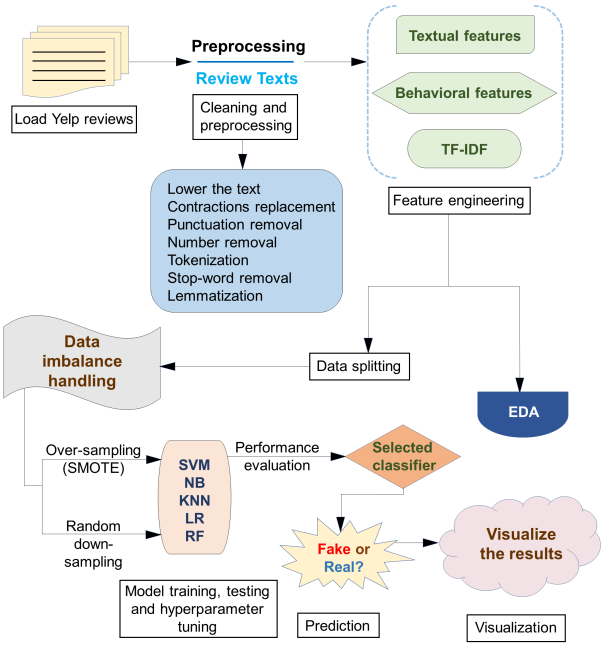

Figure 2: Overall system design

4.2.5 ***Overall system design*** Together, after preprocessing of the review texts, textual features, behavioral features and TF-IDF, features were built using YelpZip dataset. Features were filtered to reduce multicollinearity, and feature selection is conducted to improve the performance as well as reduce the computational cost. Then, data were split into training (70%) and testing (30%) datasets. Resampling techniques were applied and compared to handle the data imbalance problem. Next, SVM, NB, KNN, LR, and RF models using different combinations of features were trained and tested. The parameters of the model with best performance were further tuned. The best classifier is selected and used to predict the label of reviews from Yelp dataset. The results were then processed, manipulated and fed into an interactive spatial visualization depicting the impact of the identified fraudulent reviews on business ratings.

5 Experiments and Evaluation

The **testbed** used for model experimentation was an 8-CPu 32GB PC. The experiments were performed in Jupyter notebooks to allow for quick iteration and maximum flexibility while training and testing the models.

The questions we wanted to answer in the experiments included: 1) what were the best resampling methods to overcome the imbalance problem of dataset 2) what was the best combination of features to reduce multicollinearity and while improving model performance 3) what was the best classifier we could use for fake review identification 4) how could we best represent the results visually.

5.1 Data Set

YelpZip is a labeled data set collected from Yelp.com, it consists 608,598 reviews from 5,044 restaurants by 260,277 reviewers. In this project, YelpZip data set has been used to build the basic model for fake review detection. After training, the model was be applied on another Yelp data set of 6.99 million real reviews for 150 thousand businesses across 11 metro areas.

5.2 Handling of Imbalanced Data

The YelpZip dataset is highly imbalanced and only contains 13% of fake reviews. Models generated from training data that are highly imbalanced tend to perform poorly, and this can significantly decrease the accuracy of classification [25]. To address this problem, we had experimented with resampling methods including synthetic minority over-sampling (SMOTE), random up-sampling, and random down-sampling.

5.3 Handling of Multicollinearity

Multicollinearity is a statistical phenomenon when two or more independent features are highly correlated to each other [35]. The presence of multicollinearity may increase the standard errors of the coefficients and negatively affect the prediction of a model [36]. Figure 3 shows the correlation heatmap for both textual and behavioral features. The colors get darker as collinearity between two features increases. We've removed features 'num_word', 'num_noun', 'num_verb', 'num_adj', 'num_adv', 'num_personal_pronoun', 'typo_ratio', 'avg_word_len', 'num_positive_words','avg_user_rating', 'and positive_review_ratio' to reduce multicollinearity.

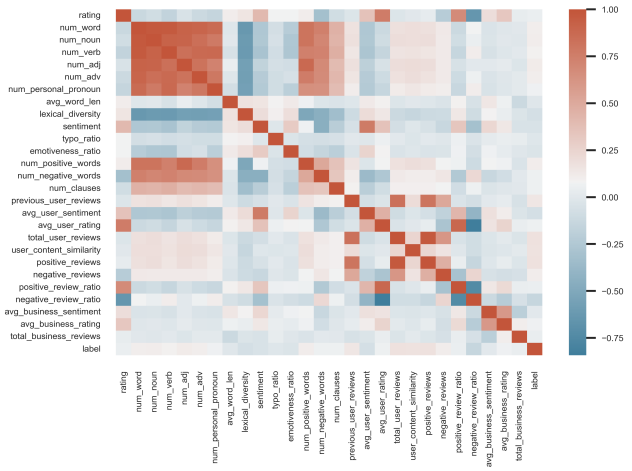

Figure 3: Correlation Heatmap for textual and behavioral features

5.4 Feature Selection

We experimented with different feature selection methods, such as forward selection and L1-based feature selection to optimize the performance of the model. Our results indicated that the combination of all textual and behavioral features (after removing multicollinearities) performed the best.

5.5 Modeling

In this set of experiments, we investigated **1)** which combination of features (all, TF-IDF only or combination of all textual and behavioral features with TF-IDF) performed the best, **2)** which resampling method was best for handling imbalanced dataset, and **3)** which model (SVM, NB, KNN, LR or RF) performed the best.

We compared the results of all combination of features, resampling techniques, and 5 supervised learning algorithms. Part of the results are shown in Table 1 due to space limitations. And the rest part of results of comparisons are shown in Appendix Table 2. The main **evaluation metrics** used to quantify the performance of classification models were accuracy, precision, recall, and F1-score.

From Table 1 and Appendix Table 2, it is evident that RF model outperformed SVM, NB, KNN and LR using the combination of all features after removing multicollinearity with TF-IDF and SMOTE as resampling technique. It showed an overall accuracy of 77%, 63% recall for the minority class, and 79% recall for the majority class in fake review detection. In addition, the comparison of textual features, behavioral features or combination with TF-IDF using linear SVM with SMOTE is shown in Appendix Table 3. Behavioral features outperformed textual features where the accuracy had increased by 18.87%. The addition of the TF-IDF features to the textual features improved accuracy from 53% to 70% using linear SVM and SMOTE, and the inclusion of behavioral features improved recall for the minority class from 45% to 64% in our baseline SVM model (Appendix Table 3).

Table 1: Results of experiments using different models and over/down-sampling techniques

Features	Sampling	Models	Overall Result			Detailed Result					
						Prec		Rec		F1	
			Acc	Prec	F1	Fake	Real	Fake	Real	Fake	Real
All	SMOTE	SVM	0.64	0.86	0.70	0.24	0.95	0.81	0.61	0.37	0.75
		RF	0.75	0.85	0.79	0.30	0.93	0.65	0.77	0.41	0.84
TF-IDF		SVM	0.75	0.82	0.78	0.25	0.90	0.45	0.80	0.32	0.85
		RF	0.64	0.82	0.70	0.20	0.91	0.57	0.65	0.30	0.76
All+TF-IDF		SVM	0.73	0.84	0.77	0.28	0.93	0.64	0.75	0.39	0.83
		RF	**0.77**	**0.85**	**0.80**	**0.31**	**0.93**	**0.63**	**0.79**	**0.41**	**0.85**
All	Random down-sampling	SVM	0.64	0.86	0.70	0.24	0.96	0.81	0.61	0.37	0.75
		RF	0.69	0.87	0.74	0.27	0.96	0.80	0.67	0.41	0.79
TF-IDF		SVM	0.65	0.83	0.70	0.22	0.93	0.67	0.64	0.33	0.76
		RF	0.56	0.83	0.62	0.19	0.93	0.74	0.53	0.31	0.67
All+TF-IDF		SVM	0.68	0.85	0.73	0.26	0.95	0.75	0.67	0.38	0.79
		RF	0.68	0.86	0.73	0.26	0.95	0.78	0.67	0.39	0.78

This table shows the comparisons of Support Vector Machine (SVM) and Random Forest (RF) model performance of all features (after removing multicollinearities), TF-IDF or a combination of all features with TF-IDF using synthetic minority over-sampling (SMOTE) or random down-sampling techniques. Evaluation metrics include accuracy (Acc), precision (Prec), F1-score (F1) and recall (Rec).

5.6 Post-Model Processing

To prepare the data for the visualization, we performed exploratory data analysis (EDA) using pandas, matplotlib, and an in-memory SAS visualization tool. This informed our pre-visualization manipulation strategy and revealed several data cleanup and transformation tasks, such as misaligned zip codes and spelling errors in address fields.

We used Python and various packages to prepare our data before loading into d3.js. For example, we used GeoPy to retrieve correct zip codes and address fields from existing latitude/longitude data. Finally, we merged our model predictions with the cleaned business and review datasets and performed aggregate calculations at the business level to be fed into the interactive visualization.

5.7 User Interface and Visualization

Throughout production, our visualization group checked in with the full group weekly to receive feedback which

resulted in experimentation of changes in data inclusion, click functionality, text, and coloring.

For visualization, we tested a range of color scales such as green to red and yellow to blue before landing on yellow to red. We made this decision after observing that red best highlighted negatively impacted areas with low or negative values and that the change in gamma provided strong distinctions across the color scale, all while accounting the potential of user color-blindness. We added a light grey overlay on the map for the selected metro area city to distinguish between urban and suburban areas. We also determined that including the chosen metro and business sector in the map title, as well as the business sector and zipcode in the bar chart title, were the best way to represent how the users choices were affecting the map and bar chart. We originally did not have without a choice for review metric and later on added in the dropdown with the addition of percent rating difference and rating change to allow for additional insights and analysis.

To improve the user interface, we added the functionality to lock the bar chart in place by clicking anywhere within the map and additionally determined that the best way to visualize this was to color the clicked zipcode in green. We also centered the map and bar chart visualizations on the page and created an empty bar chart svg upon opening the web page to keep a scroll bar whether or not the bar chart is present to account for maintaining the same layout and page structure over any range of screen and window sizes. We started with Philadelphia as the default metro and switched to Indianapolis after realizing that file size affected loading times so as to save time on the first web page loading.

6 Conclusions and discussion

Throughout the project, all team members have contributed a similar amount of effort.

It had been observed that YelpZip dataset was heavily imbalanced. Achieving a great overall accuracy didn't mean the modeling was a success. After comparing different resampling techniques, we selected SMOTE to counter the imbalance problem. The recall scores of fake and genuine reviews had reached 45% and 80% respectively in our SVM model using TF-IDF features, which were far better than the SVM model (recall for fake review: 1.23%, for real review: 99.87%) published using the same YelpZip dataset and TF-IDF features with other dynamic resampling methods [37]. Our best model showed better performance in detecting fake reviews (recall score for fake: 63%, for real: 79%), and its F1 scores (41% and 85% for fake and real reviews respectively) were better than a study used CNN model (27% and 85%) on the same YelpZip dataset [38]. To improve the overall performance of our model, we also tested different combinations of features, and found that behavioral features had better overall accuracy (63%) for the baseline model as compared to textual features (53%). This observation was consistent with results of a study published by Rastogi and Mehrotra [39]. RF with SMOTE using the combination of TF-IDF and both textual and behavioral features (accuracy: 77%) outperformed down-sampling SVM using only behavioral features (70%) on YelpZip dataset [39].

In conclusion, our random forest model using an innovative combination of textual, behavioral and word embedding features with synthetic minority over-sampling (SMOTE) proved to be a more accurate and robust model for fake review detection than previous models in the field using the same dataset.

With our visualization tool we explored the data and discovered a pattern among most cities where suburban areas were more heavily impacted by negative star rating changes and total fraudulent review counts, though the Beauty and Spa sector does not seem to follow this pattern. Automotive and Nightlife sectors have a high percentage of fraudulent reviews and rating changes with fraudulent reviews across most metropolitan areas, and the Health and Medical sector seems to have lower rating absolute difference across most metro areas as compared to other sectors in the same area. By observing the bar chart, we also noticed that across all metrics the businesses with the largest counts of fake reviews were typically not the most heavily impacted in terms of percentage of fake reviews and change in rating. In conclusion we find that we can draw more meaningful insights when incorporating additional metrics to the count of fake reviews from our model output. We also observed that when aggregating these statistics by business sector and locality, we were able to visually interpret meaningful trends in the data that can impact both consumers and business owners, alike.

References

[1] F. Salehi, B. Abdollahbeigi, A. C. Langroudi, and F. Salehi, "The impact of website information convenience on e-commerce success of companies," *Procedia-social and behavioral sciences*, vol. 57, pp. 381–387, 2012.

[2] J. Salminen, C. Kandpal, A. M. Kamel, S.-g. Jung, and B. J. Jansen, "Creating and detecting fake reviews of online products," *Journal of Retailing and Consumer Services*, vol. 64, p. 102771, 2022.

[3] H. Tang and H. Cao, "A review of research on detection of fake commodity reviews," in *Journal of Physics: Conference Series*, vol. 1651, p. 012055, IOP Publishing, 2020.

[4] X. Wang, X. Zhang, C. Jiang, and H. Liu, "Identification of fake reviews using semantic and behavioral features," in *2018 4th International Conference on Information Management (ICIM)*, pp. 92–97, IEEE, 2018.

[5] Y. Wu, E. W. Ngai, P. Wu, and C. Wu, "Fake online reviews: Literature review, synthesis, and directions for future research," *Decision Support Systems*, vol. 132, p. 113280, 2020.

[6] S. He, B. Hollenbeck, and D. Proserpio, "The market for fake reviews," *Marketing Science*, vol. 41, no. 5, pp. 896–921, 2022.

[7] T. Lappas, G. Sabnis, and G. Valkanas, "The impact of fake reviews on online visibility: A vulnerability assessment of the hotel industry," *Information Systems Research*, vol. 27, no. 4, pp. 940–961, 2016.

[8] M. Luca and G. Zervas, "Fake it till you make it: Reputation, competition, and yelp review fraud," *Management Science*, vol. 62, no. 12, pp. 3412–3427, 2016.

[9] A. Kumar, R. D. Gopal, R. Shankar, and K. H. Tan, "Fraudulent review detection model focusing on emotional expressions and explicit aspects: investigating the potential of feature engineering," *Decision Support Systems*, vol. 155, p. 113728, 2022.

[10] R. Mohawesh, S. Xu, S. N. Tran, R. Ollington, M. Springer, Y. Jararweh, and S. Maqsood, "Fake reviews detection: A survey," *IEEE Access*, vol. 9, pp. 65771–65802, 2021.

[11] A. Ghose and P. G. Ipeirotis, "Estimating the helpfulness and economic impact of product reviews: Mining text and reviewer characteristics," *IEEE Transactions on Knowledge and Data Engineering*, vol. 23, no. 10, pp. 1498–1512, 2011.

[12] M. Rahman, B. Carbunar, J. Ballesteros, and D. H. P. Chau, "To catch a fake: Curbing deceptive yelp ratings and venues," *Statistical Analysis and Data Mining: The ASA Data Science Journal*, vol. 8, no. 3, pp. 147–161, 2015.

[13] A. S. Md. Tayeen, A. Mtibaa, and S. Misra, "Location, location, location! quantifying the true impact of location on business reviews using a yelp dataset," in *2019 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*, pp. 1081–1088, 2019.

[14] C. Fu, M. Zhao, L. Fan, X. Chen, J. Chen, Z. Wu, Y. Xia, and Q. Xuan, "Link weight prediction using supervised learning methods and its application to yelp layered network," *IEEE Transactions on Knowledge and Data Engineering*, vol. 30, no. 8, pp. 1507–1518, 2018.

[15] W. N. Dilla and R. L. Raschke, "Data visualization for fraud detection: Practice implications and a call for future research," *International Journal of Accounting Information Systems*, vol. 16, pp. 1–22, 2015.

[16] C. G. Harris, "Detecting fake yelp reviews using a positional lstm / k-l divergence ensemble approach," in *2022 1st International Conference on Information System & Information Technology (ICISIT)*, pp. 61–66, 2022.

[17] E. Elmurngi and A. Gherbi, "An empirical study on detecting fake reviews using machine learning techniques," pp. 107–114, 08 2017.

[18] H. Tufail, M. U. Ashraf, K. Alsubhi, and H. M. Aljahdali, "The effect of fake reviews on e-commerce during and after covid-19 pandemic: Skl-based fake reviews detection," *IEEE Access*, vol. 10, pp. 25555–25564, 2022.

[19] R. Agarwal and D. K. Sharma, "Detecting fake reviews using machine learning techniques: a survey," in *2022 2nd International Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE)*, (Greater Noida, India), pp. 1750–1756, 2022.

[20] J. Salminen, C. Kandpal, A. M. Kamel, S. gyo Jung, and B. J. Jansen, "Creating and detecting fake reviews of online products," *Journal of Retailing and Consumer Services*, vol. 64, p. 102771, 2022.

[21] R. Mohawesh, S. Xu, M. Springer, M. Al-Hawawreh, and S. Maqsood, "Fake or genuine? contextualised text representation for fake review detection," in *Natural Language Processing*, Academy and Industry Research Collaboration Center (AIRCC), dec 2021.

[22] F. W. A. Mosleh Hmoud Al-Adhaileh, "Detecting and analysing fake opinions using artificial intelligence algorithms," *Intelligent Automation & Soft Computing*, vol. 32, no. 1, pp. 643–655, 2022.

[23] J. Wang, R. Wen, C. Wu, Y. Huang, and J. Xion, "Fdgars: Fraudster detection via graph convolutional networks in online app review system," pp. 310–316, 05 2019.

[24] S. Nilizadeh, H. Aghakhani, E. Gustafson, C. Kruegel, and G. Vigna, "Think outside the dataset: Finding fraudulent reviews using cross-dataset analysis," pp. 3108–3115, 05 2019.

[25] C. G. Harris, "Detecting fraudulent online yelp reviews using k-l divergence and linguistic features," *Procedia Computer Science*, vol. 204, pp. 618–626, 2022. International Conference on Industry Sciences and Computer Science Innovation.

[26] J. K. Rout, A. Dalmia, K.-K. R. Choo, S. Bakshi, and S. K. Jena, "Revisiting semi-supervised learning for online deceptive review detection," *IEEE Access*, vol. 5, pp. 1319–1327, 2017.

[27] C.-C. Chang and C.-J. Lin, "Libsvm: a library for support vector machines," *ACM transactions on intelligent systems and technology (TIST)*, vol. 2, no. 3, pp. 1–27, 2011.

[28] G. I. Webb, E. Keogh, and R. Miikkulainen, "Naïve bayes.," *Encyclopedia of machine learning*, vol. 15, pp. 713–714, 2010.

[29] D. Berrar, "Bayes' theorem and naive bayes classifier," *Encyclopedia of Bioinformatics and Computational Biology: ABC of Bioinformatics*, vol. 403, p. 412, 2018.

[30] A. M. Elmogy, U. Tariq, M. Ammar, and A. Ibrahim, "Fake reviews detection using supervised machine learning," *International Journal of Advanced Computer Science and Applications*, vol. 12, no. 1, 2021.

[31] Q. Kuang and L. Zhao, "A practical gpu based knn algorithm," in *Proceedings. The 2009 International Symposium on Computer Science and Computational Technology (ISCSCI 2009)*, p. 151, Citeseer, 2009.

[32] M. Pal, "Random forest classifier for remote sensing classification," *International journal of remote sensing*, vol. 26, no. 1, pp. 217–222, 2005.

[33] L. Breiman, "Random forests," *Machine learning*, vol. 45, pp. 5–32, 2001.

[34] S. N. Alsubari, S. N. Deshmukh, A. A. Alqarni, N. Alsharif, T. Aldhyani, F. W. Alsaade, and O. I. Khalaf, "Data analytics for the identification of fake reviews using supervised learning," *Computers, Materials & Continua*, vol. 70, no. 2, pp. 3189–3204, 2022.

[35] H. Midi, S. K. Sarkar, and S. Rana, "Collinearity diagnostics of binary logistic regression model," *Journal of interdisciplinary mathematics*, vol. 13, no. 3, pp. 253–267, 2010.

[36] G. R. Franke, "Multicollinearity," *Wiley international encyclopedia of marketing*, 2010.

[37] G. S. Budhi, R. Chiong, and Z. Wang, "Resampling imbalanced data to detect fake reviews using machine learning classifiers and textual-based features," *Multimedia Tools and Applications*, vol. 80, pp. 13079–13097, 2021.

[38] Q. Li, Q. Wu, C. Zhu, J. Zhang, and W. Zhao, "An inferable representation learning for fraud review detection with cold-start problem," in *2019 international joint conference on neural networks (IJCNN)*, pp. 1–8, IEEE, 2019.

[39] A. Rastogi and M. Mehrotra, "Impact of behavioral and textual features on opinion spam detection," in *2018 Second International Conference on Intelligent Computing and Control Systems (ICICCS)*, pp. 852–857, IEEE, 2018.

A Appendix

Features	Sampling	Models	Overall Result			Detailed Result					
						Prec		Rec		F1	
			Acc	Prec	F1	Fake	Real	Fake	Real	Fake	Real
All	SMOTE	NB	0.54	0.87	0.61	0.21	0.97	0.88	0.49	0.34	0.65
		KNN	0.70	0.83	0.75	0.24	0.92	0.58	0.72	0.34	0.81
		LR	0.64	0.86	0.70	0.24	0.96	0.81	0.62	0.38	0.75
TF-IDF		NB	0.74	0.83	0.77	0.26	0.92	0.53	0.77	0.35	0.84
		LR	0.75	0.82	0.78	0.26	0.91	0.46	0.80	0.33	0.85
		KNN	Execution time not suitable for large dataset								
All+TF-IDF		NB	0.76	0.79	0.77	0.19	0.88	0.25	0.83	0.22	0.86
		LR	0.70	0.86	0.75	0.28	0.95	0.76	0.70	0.40	0.80
		KNN	Execution time not suitable for large dataset								
All	down-sampling	NB	0.54	0.87	0.61	0.21	0.97	0.89	0.49	0.34	0.65
		KNN	0.67	0.85	0.72	0.24	0.94	0.73	0.66	0.37	0.77
		LR	0.64	0.86	0.70	0.24	0.96	0.81	0.62	0.37	0.75
TF-IDF		NB	0.53	0.84	0.60	0.19	0.93	0.77	0.49	0.30	0.64
		KNN	0.21	0.51	0.19	0.13	0.88	0.91	0.10	0.23	0.18
		LR	0.66	0.84	0.72	0.23	0.93	0.69	0.66	0.35	0.77
All+TF-IDF		NB	0.57	0.85	0.64	0.21	0.94	0.78	0.54	0.33	0.69
		KNN	0.67	0.85	0.72	0.25	0.94	0.73	0.66	0.37	0.77
		LR	0.69	0.86	0.74	0.27	0.95	0.78	0.68	0.40	0.79
All	Random over-sampling	SVM	0.73	0.85	0.77	0.27	0.93	0.65	0.74	0.38	0.82
		NB	0.54	0.87	0.61	0.21	0.97	0.89	0.49	0.34	0.65
		LR	0.70	0.86	0.74	0.27	0.95	0.77	0.68	0.40	0.80
		KNN	0.71	0.83	0.75	0.24	0.92	0.56	0.74	0.34	0.82
		RF	0.66	0.85	0.72	0.25	0.95	0.76	0.65	0.37	0.77
TF-IDF		SVM	0.70	0.83	0.74	0.24	0.92	0.58	0.72	0.34	0.80
		NB	0.69	0.83	0.74	0.24	0.92	0.61	0.70	0.34	0.80
		KNN	Execution time not suitable for large dataset								
		LR	0.69	0.84	0.74	0.24	0.93	0.63	0.70	0.35	0.80
		RF	0.59	0.83	0.65	0.20	0.93	0.71	0.57	0.31	0.71
All+TF-IDF		SVM	0.72	0.85	0.76	0.27	0.93	0.65	0.74	0.39	0.82
		NB	0.58	0.84	0.65	0.21	0.94	0.77	0.55	0.33	0.70
		KNN	Execution time not suitable for large dataset								
		LR	0.70	0.86	0.75	0.27	0.95	0.77	0.69	0.40	0.80
		RF	0.70	0.86	0.74	0.27	0.95	0.76	0.69	0.40	0.80

Table 2: Results of experiments using different combinations of features, models and resampling methods

Features	Overall Result			Detailed Result					
				Prec		Rec		F1	
	Acc	Prec	F1	Fake	Real	Fake	Real	Fake	Real
Textual	0.53	0.80	0.60	0.16	0.90	0.61	0.52	0.25	0.65
Behavioral	0.63	0.86	0.69	0.24	0.96	0.82	0.60	0.37	0.74
Textual+TF-IDF	0.70	0.83	0.75	0.24	0.92	0.56	0.73	0.34	0.81
Behavioral+TF-IDF	0.73	0.84	0.77	0.27	0.93	0.64	0.74	0.38	0.83

Table 3: Model performance of linear SVM using different features with SMOTE resampling method