Vasudha Jasthi, Gaurav Shinde

Fraudulent Job Posting Classifier and Job Posting Clustering

Introduction:

With the pandemic and the advent and ease of technology, many more companies today than ever are posting their job vacancies online on various websites. Even though this is very convenient for companies, this poses a risk of job scams and fraudulent job postings known as Online Recruitment Fraud (ORF). This is a new challenge to both companies and job-seekers because it leads to stolen money, and job-seekers having false hopes in getting employed. According to the Better Business Bureau, "Employment scams were the #1 riskiest scam in both 2018 and 2019… An estimated 14 million people are exposed to employment scams with more than $2 billion lost per year, not counting time or emotional losses. The risk of this scam continues to rise…".

For every one of us, finding a job is an unavoidable task. According to TopResume, the average job-hunting time in the United States takes about 5 months. 5 months is a long time when thinking in terms of finances and daily life for instance. The general steps of job finding includes sifting through numerous websites with job ads, reading through each job description and features, applying to the jobs, etc. Overall, the entire process is a tiring and a laborious one, especially to read every description in order to figure out if it matches the job seekers skills and wants.

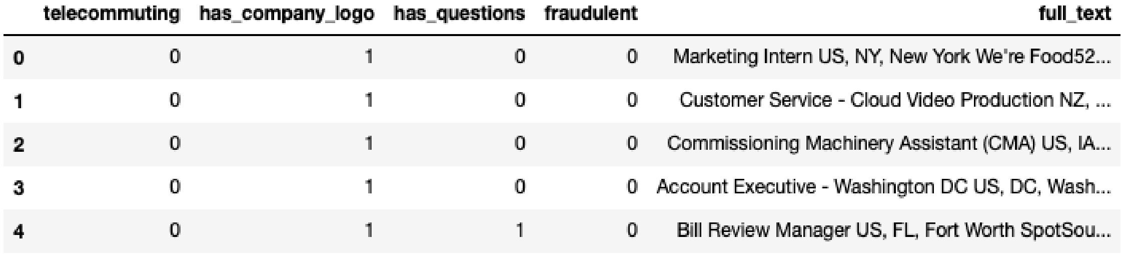
This project seeks to help these problems by building a model to determine whether a job ad posting is fraudulent, based on recent job ad postings, and creating a model to identify the most similar job descriptions. The "Fraudulent Job Posting Classifier" determines whether a particular job ad is fake or not using features of a job posting such as the description of the ad.

Approach:

Before creating our models, we will perform various Exploratory Data Analysis methods and techniques so we can understand the data better. We will explore what are the most common words used in the postings as well as any high frequency groups among features fraudulent vs non fraudulent.

We will be using the Kaggle dataset Fake JobPostings which has information about various jobs including descriptions and whether they are fraudulent or not. This data set had 17,880 number of job posts. We will use this dataset for all of our analyses and models in the project. Before we start with building models, since the dataset is unbalanced, we will need to use imblearn for oversampling techniques. We will also then use data preprocessing methods for any missing values, stopword and unnecessary characters removal, and feature vectorizing (ex. Tf_idf vectorization). For the classification of fake job postings, we will be splitting the dataset into training, validation, testing sets to create the model. We plan to use an ensemble model of logistic regression, k-nearest neighbor, random forest classifier, and multinomial naive bayes models. GridSearchCV will also be used to find the best hyperparameters and we will use f1 score, accuracy, precision, and recall metrics to compare models and choose the best ones. We will finally choose the top 3 models from the models created and construct a Voting Classifier model which is an ensemble model technique. Included below is a step by step road map to create the classification model:

1. Preprocessing:
    a. Missing Values: removed any features that contained 50% or more NaNs.
    b. Any other missing text feature rows, replace NaN with ‘’ (empty character).
    c. Combined all the Text columns into a new feature.
    d. Dropped the combined features.



	telecommuting	has_company_logo	has_questions	fraudulent	full_text
0	0	1	0	0	Marketing Intern US, NY, New York We're Food52...
1	0	1	0	0	Customer Service - Cloud Video Production NZ, ...
2	0	1	0	0	Commissioning Machinery Assistant (CMA) US, IA...
3	0	1	0	0	Account Executive - Washington DC US, DC, Wash...
4	0	1	1	0	Bill Review Manager US, FL, Fort Worth SpotSou...

*This is how the data looks after combining the text column

    e. Removed punctuation marks, brackets html, urls, and any other unnecessary characters in the full_text column.
    f. Tokenize the full_text column.
    g. Apply stemming to each word in the column.
    h. Remove stop words .
2. Apply TFIDF Vectorizer and Countvectorizer (Bag of Words) and combine the vectors into 2 different datasets (df_tfidf, df_cv).
3. Balance both the datasets with oversampling method SMOTE.
4. Train and split the datasets using the 80:20 rule.
5. Create the various models and computer confusion matrix with target = fraudulent feature and the x = all other columns:
    a. Logistic Regression - tfidf
    b. Logistics Regression - BOW
    c. KNN - tfidf
    d. KNN - BOW
    e. MLP - tfidf
    f. MLP - BOW
    g. Random Forest Classifier - tfidf
    h. Random Forest Classifier - BOW
6. Choose the Top3 performing models from the 8 created above and run a Voting Classifier model (both hard and soft and for each tfidf and bow to compare)

Results:

We created many graphs during the exploring the data part. In Figure 1. we can see that the data is imbalanced as there are significantly more non-fraudulent cases than fraudulent cases. This is also most likely just a natural phenomenon. According to Figure 2, the largest % of fraud jobs fall in the full-time category. Figure 3 shows that the most amount of fraud jobs called for a certification. Figure 4 shows the % of fraud jobs by telecommuting. We hypothesize that telecommuting convenience may make employees overlook other aspects about the job posting, which may be a technique used by fraud job postings. We hypothesize that function may make employees overlook other aspects about the job posting, which may be a technique used by fraud job postings. We constructed a few more graphs, but we also created a correlation heat map matrix, as shown in Figure 5, between all of the features with the target feature which is the ’fraudulent’ column. Even though not many of the features are highly correlated with the target, we will still include it in our models to see how they will affect them. We also created word clouds to see what are the most common words among fraudulent job postings and non-fraudulent job postings, as shown in Figure 6 and 7. In both, the phrase full time appeared frequently. Words such as bachelor's degree, and others relating to having more experience appeared more in non-fraudulent; words like highschool and product/customer shows up more in fraudulent postings that could be designed for entry level job luring.

For the Classification Model, we created 8 models plus 4 Voting classifier models with the top 3 models/the 8 created. Here is a table summarizing the metrics:

Model	Accuracy	Precision	Recall	F1 Score
Log-Reg TFIDF	92%	90%	93%	92%
Log-Reg BOW	95%	94%	96%	95%
KNN (k = 2) TFIDF	87%	88%	85%	87%
KNN (k = 2) BOW	91%	88%	96%	92%
MLP TFIDF	51%	50%	100%	67%
MLP BOW	68%	68%	68%	68%
Random Forest Classifier - TFIDF	99%	99%	99%	99%
Random Forest Classifier - BOW	79%	100%	60%	75%

From this table, we can see that the Random Forest Classifier with tfidf performed the best with accuracy 99% out of all the eight models created and the least performing is the MLP tfidf model with accuracy about 51%.

In this table below, we display the metrics for the Voting Classifiers that we ran when combining Logistic Regression, K-Nearest Neighbors, and Random Forest Classifier Models.

Model	Accuracy	Precision	Recall	F1 Score
Voting Classifier - TFIDF (hard)	98%	98%	98%	98%

Voting Classifier - TFIDF (soft)	97%	97%	98%	97%
Voting Classifier - BOW (hard)	98%	98%	99%	99%
Voting Classifier - BOW (soft)	98%	97%	100%	98%

Overall, all four of the classification models above did really well with each having at least a 97% accuracy.

Conclusion:

The main goal of this project was to identify key traits of job descriptions which are fraudulent in nature, to create a classification model that detects fraudulent job postings and to cluster jobs based on descriptions.

For the classification model, we preprocessed the data, balanced the data using SMOTE, ran Logistic regression, K-Nearest Neighbors, MultiLayer Perceptron, and Random Forest Classifier models for each tfidf vectorizer and countvectorizer. For each model, we performed GridSearchCV to choose the best parameters possible for each model. We found the top 3 models to be Random Forest Classifier, Soft Ensemble Voting, and Hard Ensemble Voting with the highest accuracy, precision, recall, and f1-score according to the table.

Acknowledgements:

- https://philpapers.org/archive/NASMLA-3.pdf
- https://dorsam7.medium.com/using-word-embedding-to-build-a-job-search-engine-a43a5d443009
- https://www.researchgate.net/publication/318814319_Skill2vec_Machine_Learning_Approaches_for_Determining_the_Relevant_Skill_from_Job_Description
- https://www.bbb.org/globalassets/local-bbbs/council-113/media/bbb-institute/employmentscams/2020-bbb-employmentscams-report.pdf
- https://www.kaggle.com/shivamb/real-or-fake-fake-jobposting-prediction/code
- sklearn.ensemble.VotingClassifier — scikit-learn 0.24.2 ...http://scikit-learn.org › stable › modules › generated › s…
- SMOTE for Imbalanced Classification with Pythonhttps://machinelearningmastery.com › Blog
- Python | Stemming words with NLTK - GeeksforGeekshttps://www.geeksforgeeks.org › python-stemming-wor…
- https://blog.paperspace.com/implementing-levenshtein-distance-word-autocomplete-autocorrect/#:~:text=The%20Levenshtein%20distance%20is%20a,transform%20one%20word%20into%20another.
- https://towardsdatascience.com/cheat-sheet-to-implementing-7-methods-for-selecting-optimal-number-of-clusters-in-python-898241e1d6ad

- https://smorbieu.gitlab.io/accuracy-from-classification-to-clustering-evaluation/#:~:text=Computing%20accuracy%20for%20clustering%20can,the%20accuracy%20for%20clustering%20results.
- https://link.springer.com/article/10.1007/s40595-016-0086-9
- https://validclust.readthedocs.io/en/latest/validclust.html
- https://dzone.com/articles/kmeans-silhouette-score-explained-with-python-exam
- https://people.csail.mit.edu/dsontag/courses/ml14/notes/Luxburg07_tutorial_spectral_clustering.pdf

Figures from Exploratory Data Analysis explained earlier in Results:

Figure 1:

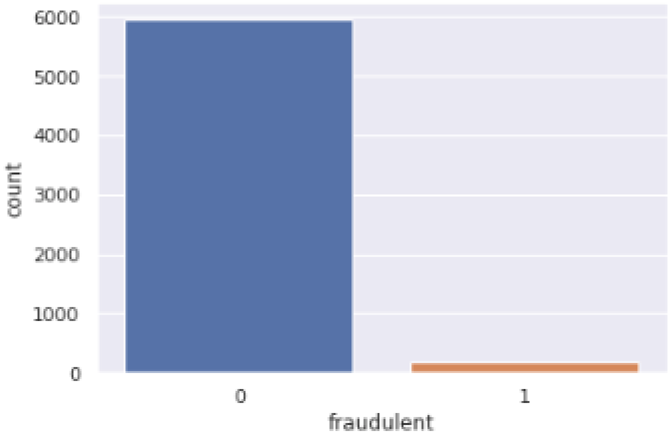


Figure 2:

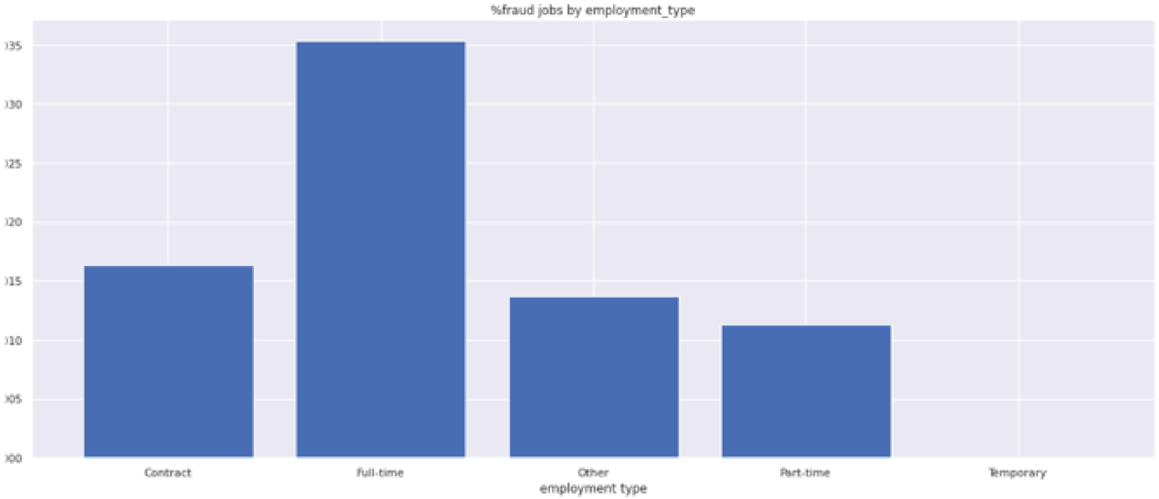
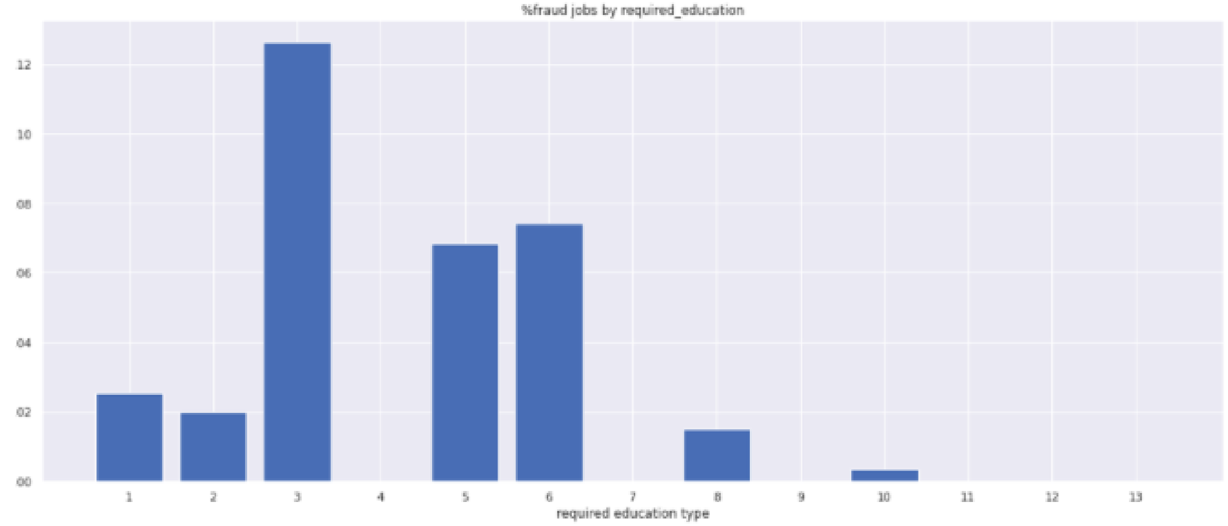


Figure 3:



* index 1: Associate Degree, index 2: Bachelor's Degree, index 3: Certification, index 4: Doctorate, index 5: High School or equivalent, index 6: Master's Degree, index 7: Professional, index 8: Some College Coursework Completed, index 9: Some High School Coursework, index 10: Unspecified, index 11: Vocational, index 12: Vocational - Degree, index 13: Vocational - HS Diploma,

Figure 4:

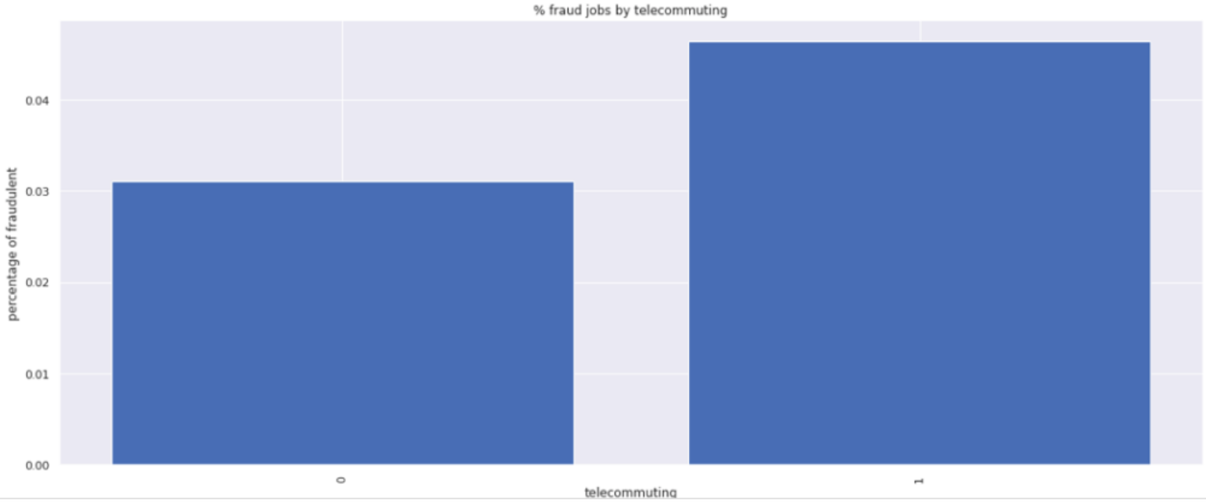


Figure 5:

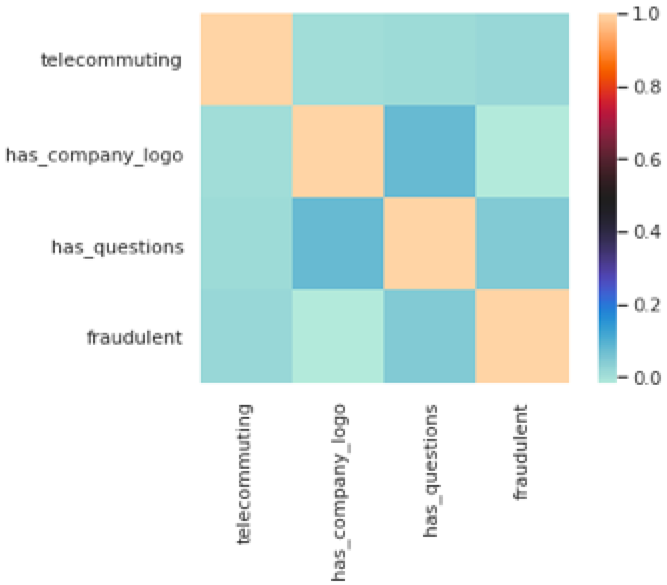


Figure 6. Non-Fraudulent Jobs

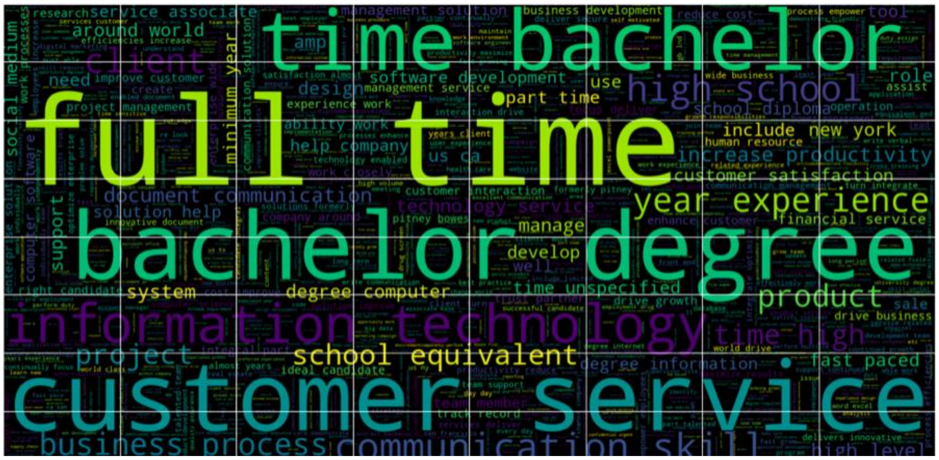
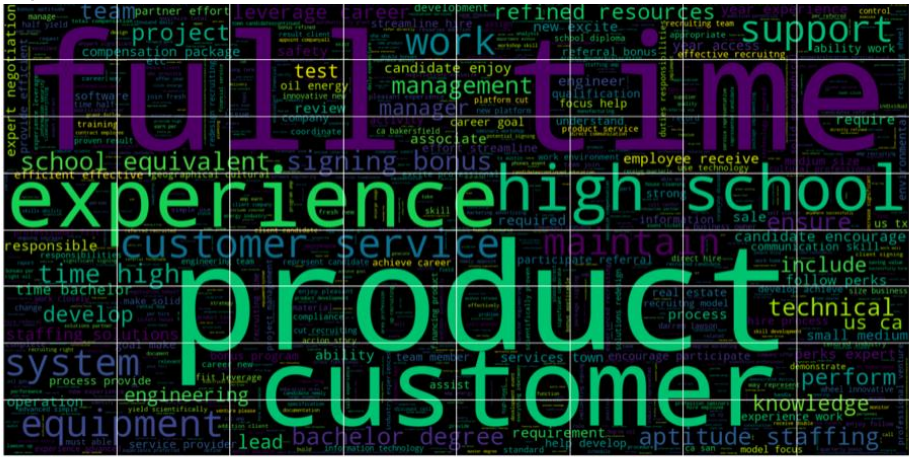


Figure 7. Fraudulent Jobs


full time
product
customer
experience
high school
customer service
school equivalent
signing bonus
support
work
management
project
system
equipment
bachelor degree
aptitude
staffing
technical
knowledge
perform
include
refined resources
maintain
time high
develop
engineering