Recognising fraudulent job postings

...

Group 10
(Aleksandra Seregina, Tomi Haapakoski,
Niels van Slooten, Linh Hoang)

Business context: why to analyse fraudulent job postings?

- According to a 2021 Microsoft survey, 40% of employees are actively considering leaving their jobs. McKinsey & Co (2021) discovered that up to 25% more workers are willing to switch occupations due to accelerated trends in remote working in the post-pandemic world.

- This trend led to an increase in job scams. Kathryn Vasel (2022) in her article for CNN claims that Federal Trade Commission's division in 2021 received twice more complaints about job scams than in 2020. Vasel states that scammers use different tactics: some try to gain access to personal information, while others might solicit payments or hire for an illegal task.

- Commonly, users get scammed through a published job posting. As students, we are interested to see whether described challenge can be addressed through automated classification of job postings. We believe that this study is beneficial for job-seekers and for recruiting companies struggling with efficient job posting prescreening.

Aims of the study: what questions does the study answer?

Main goal: assess feasibility of accurate automatic classification of fraudulent job postings.

H1: Fraudulent job postings can be identified with >70% accuracy through classification models, bag-of-words models or a combination of them

---

Secondary goal: identify relevant predictors among the attributes available in the dataset.

H2: Features such as salary range, information about the employer, required experience and education, and combinations of those have the strongest correlation with fraudness.

Methods: Majority of features are categorical, including many free text ones. Hence, the study will be conducted in 2 parts: first, classification based on primary attributes will be done through decision tree and random forest models; second, free text features will be transformed into bag-of-words dataset that will be analysed through naive bayes and logistics regression models. Based on the results, the best model will be chosen.

Exploratory Data Analysis: Data set and distribution

Shape: 17880 rows and 18 columns (17880, 18)

The features of the dataset can be found below, there are 5 features with the datatype integer. 13 features have datatype object. What also stands out is the presence of many of null values.

```
 #   Column               Non-Null Count  Dtype
---  ------               --------------  -----
 0   job_id               17880 non-null  int64
 1   title                17880 non-null  object
 2   location             17534 non-null  object
 3   department           6333 non-null   object
 4   salary_range         2868 non-null   object
 5   company_profile      14572 non-null  object
 6   description          17879 non-null  object
 7   requirements         15185 non-null  object
 8   benefits             10670 non-null  object
 9   telecommuting        17880 non-null  int64
 10  has_company_logo     17880 non-null  int64
 11  has_questions        17880 non-null  int64
 12  employment_type      14409 non-null  object
 13  required_experience  10830 non-null  object
 14  required_education   9775 non-null   object
 15  industry             12977 non-null  object
 16  function             11425 non-null  object
 17  fraudulent           17880 non-null  int64
dtypes: int64(5), object(13)
```

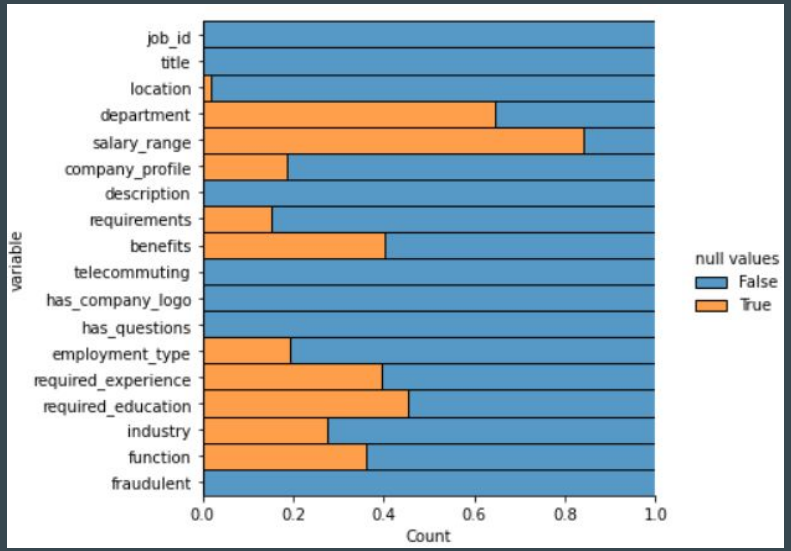
Figure below shows that the data is really imbalanced. 95.2% of the data is classified as no fraud, whereas only 4.8% is classified as fraudulent. This problem needs to be tackled when analyzing the data.

Distribution of fraudulent vs non fraudulent ads

Exploratory Data Analysis: Visualization of null vs non-null values

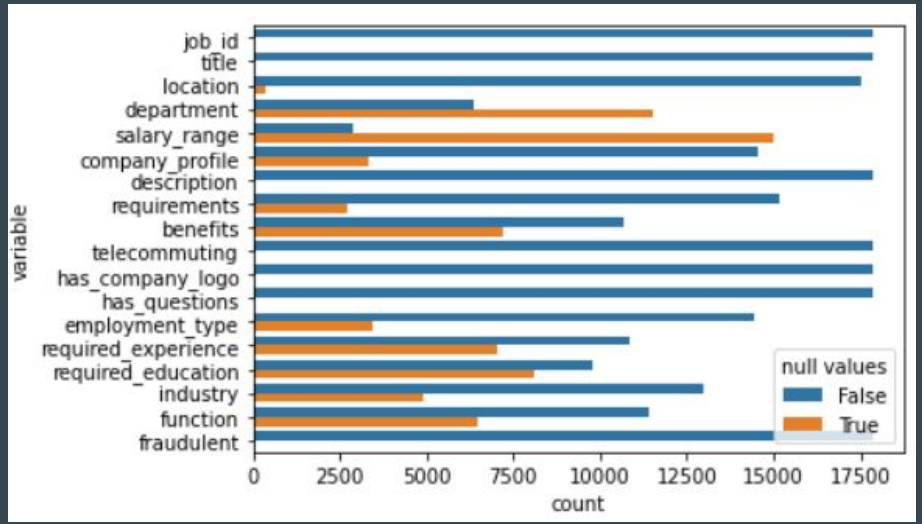
Null: 11 out of 18 features contained null values.

Distribution of null vs non-null values per feature



High count > 50% null: salary_range & department

Count of null vs non null values per feature



Exploratory Data Analysis: Deep dive numerical features

In the figure below, the correlations between the numerical features are shown. One value that stands out is the **-0.262** for the correlation between **has_company_logo and fraudulent** ads. This indicates that the presence of a company logo within an ad reduces the probability of the advertisement being fraudulent.

Correlations between numerical features

	job_id	telecommuting	has_company_logo	has_questions	fraudulent
job_id	1.000000	-0.004559	-0.014539	-0.087025	0.079872
telecommuting	-0.004559	1.000000	-0.019836	0.020345	0.034523
has_company_logo	-0.014539	-0.019836	1.000000	0.233932	-0.261971
has_questions	-0.087025	0.020345	0.233932	1.000000	-0.091627
fraudulent	0.079872	0.034523	-0.261971	-0.091627	1.000000

In the figure below, you can see that when there is no logo present, there is a **15.9%** probability of being a fraudulent ad. Whereas there is only a **2.0%** probability when there is a logo present.

Distribution of fraudulent vs non fraudulent ads by presence of a company logo

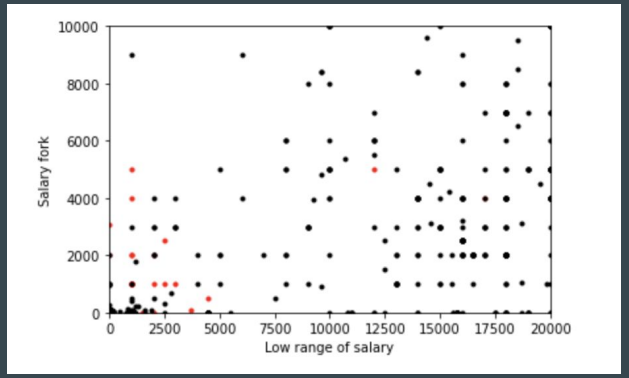
Classification based on primary attributes: feature engineering

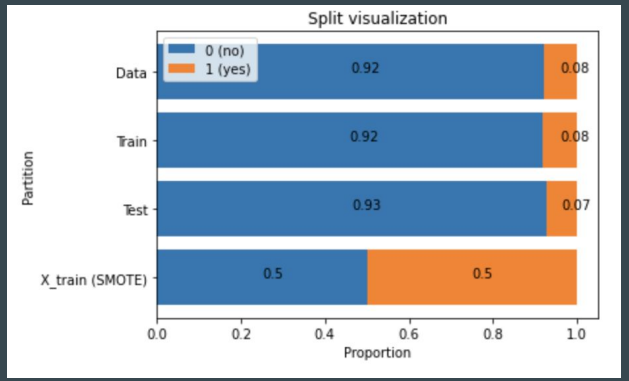
This table shows features that were removed or transformed for classification based on primary attributes. Features not shown here were unchanged. Categorical features were encoded through get_dummies.

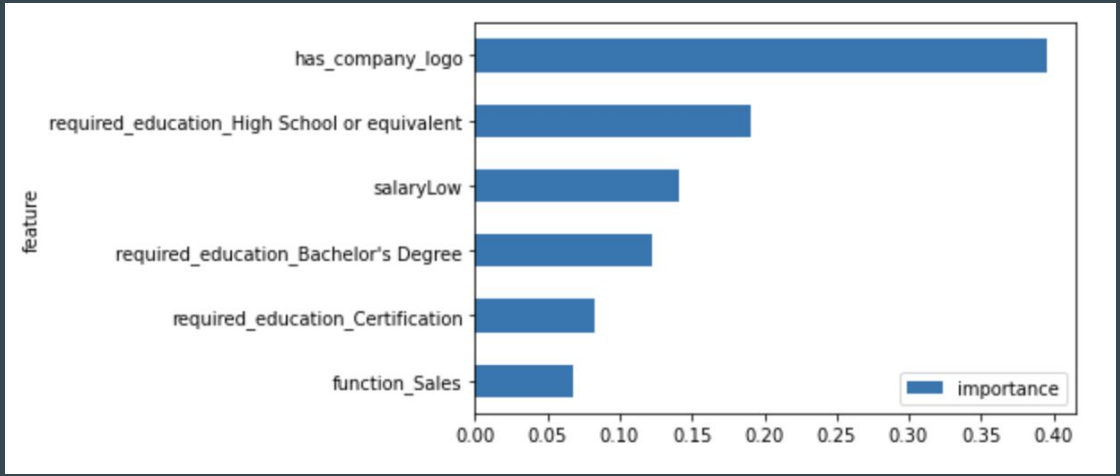
Removed	‘job_id’, ‘description’, ‘title’	Either irrelevant OR free text with no null values to be transformed into binary feature
Converted to numeric	‘salary_range’	Converted to numeric and parsed into ‘salary low margin’ and ‘salary high margin features’
Parsed and reduced number of categories	‘location’	Country code was retrieved into a separate feature; Countries with highest frequency were unchanged and the rest were transformed to ‘Others’
Reduced number of categories	‘department’, ‘industry’, ‘function’	Most frequent categories were unchanged and the rest were transformed to ‘Others’ category
Transformed to binary variables	‘company_profile’, ‘requirements’, ‘benefits’	Transformed to binary features where 1 = has data and 0 = no data available (null)

Classification based on primary attributes: data preparation

- After feature modifications, there were still null values for salaries. Salary was assumed to be top predictor, as frauduness tend to occur in job postings with low salary. Hence null values were removed from the dataset.
- Final dataset included 2841 observations.
- Train:test split was done in proportion 70:30.
- Data was rebalanced with SMOTE. This method was preferred over undersampling to avoid losing observations that would help to answer H2.
- Testing was done on the test data!



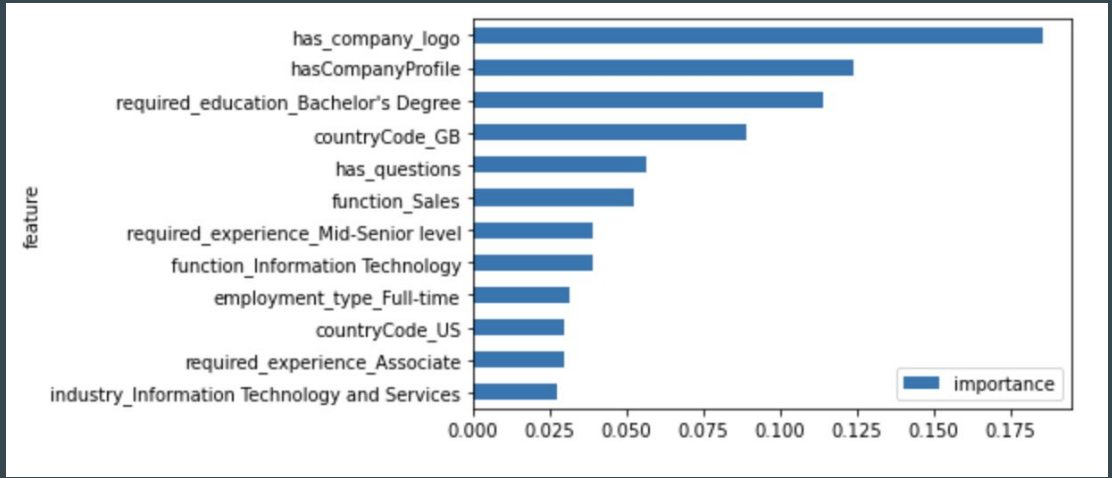




Decision tree

Accuracy score: 82.53

Features with importance > 0



Random forest

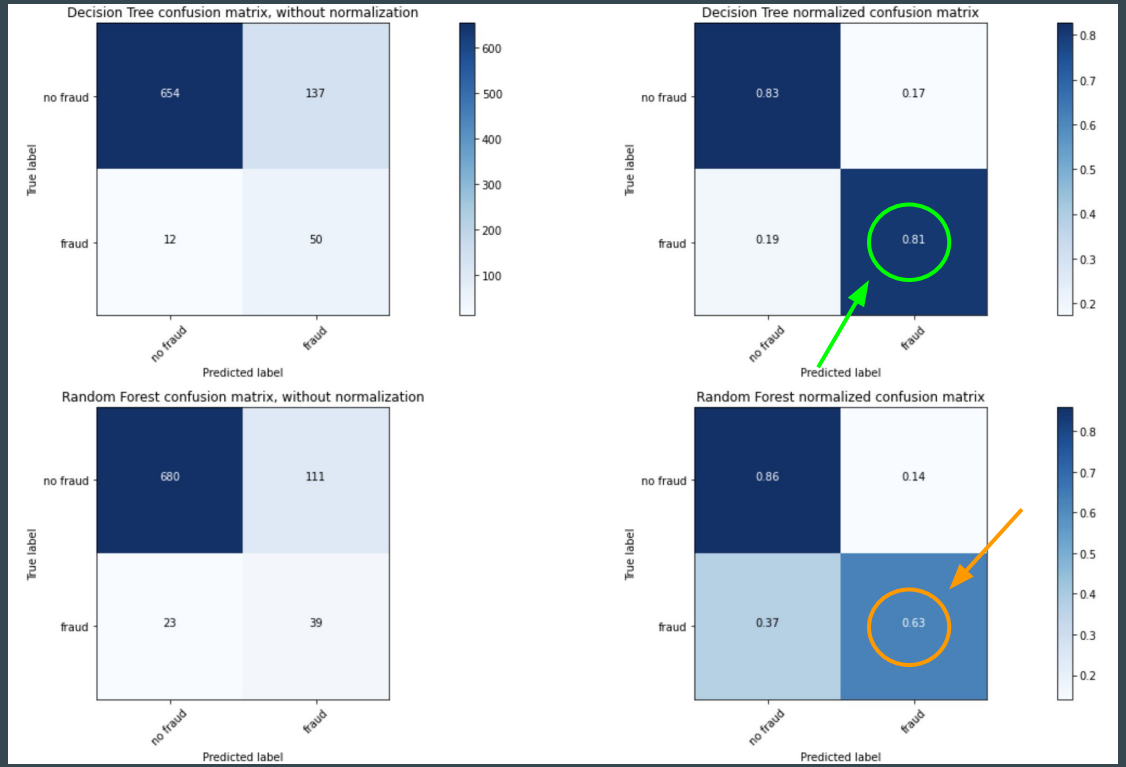
Accuracy score: 84.29

Features with importance > 0.025

Model comparison: Decision tree outperforms random forest

- Compared to decision tree, random forest has higher overall accuracy (84.29% vs 82.52%) and higher accuracy in identifying non fraudulent job postings (86% vs 83%).

- Nevertheless, in business context it is more crucial to identify better fraudulent observations. Hence, decision tree outperforms random forest with TN rate of 81% compared to random forest TN rate of 63%.

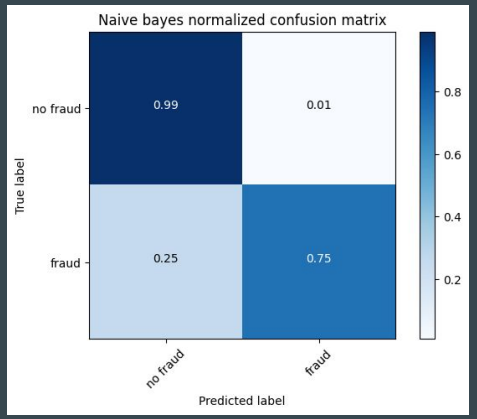


Bag-of-words models: context and data preparation

- Most of the dataset is in free text format which is hard to label automatically. However, leaving out the free text variables from the models signifies that the models are trained with small subset of available data.
- To ensure that any relevant data is not lost by leaving out free text features, we are also training a pair of models with bag-of-words of the job postings.
- Bag-of-words is essentially a list of tuples containing individual words and the count of their appearances. However, bag-of-words are often long and hard to compute with. That implies they should not be used in case they do not offer significantly better performance when compared to previously introduced models.

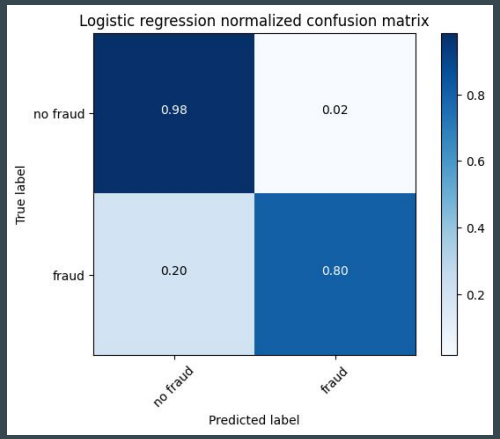
- Bag-of-words models were trained with SMOTE re-balanced dataset using train:test split in 70:30 proportion. Testing was done on the test data.

Bag-of-words models: naive Bayes and logistic regression



Naive Bayes

Accuracy score: 97.86

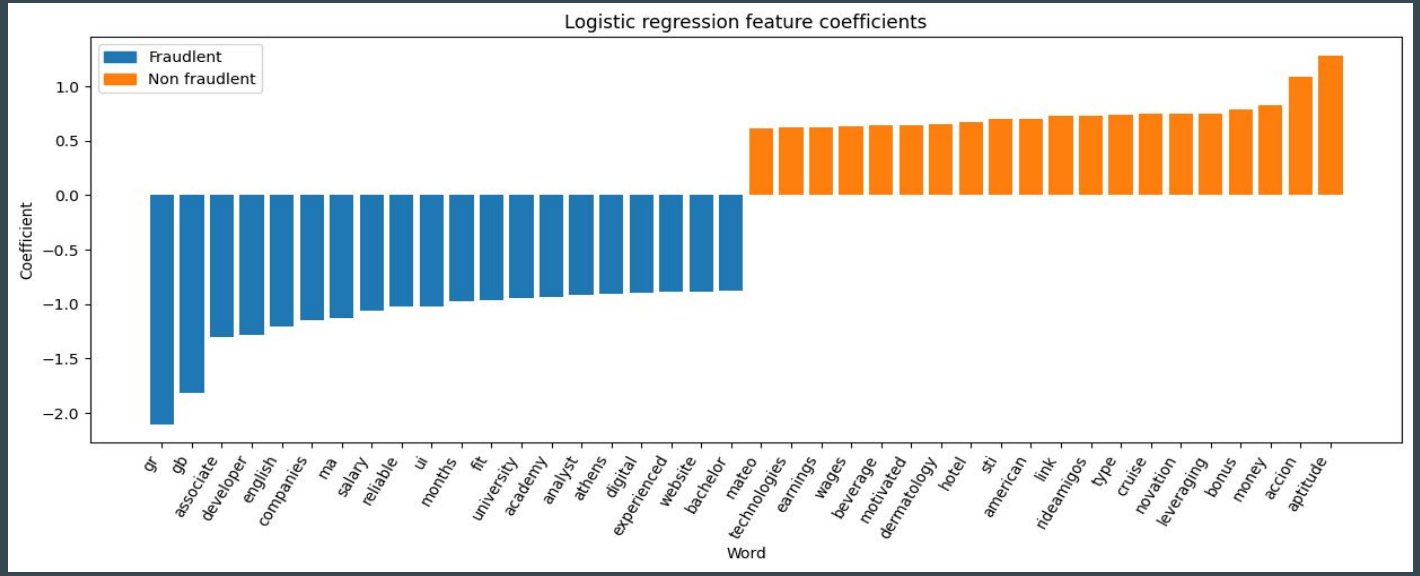


Logistic regression

Accuracy score: 97.56

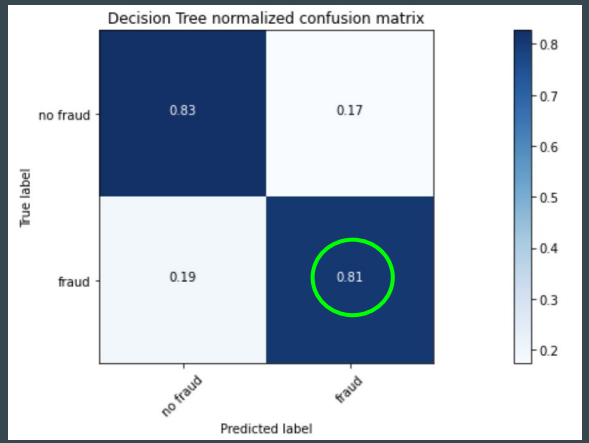
Bag-of-words models: feature importance

Models trained with bag-of-words are most accurate when identifying real job postings. However, they do not perform better with fraudulent job postings. Most significant coefficients for fraudulent job postings appear to be country codes, which are already included in the previous models

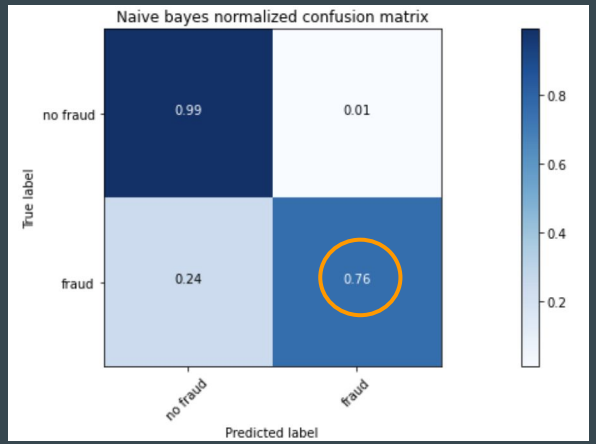


Model comparison: decision tree vs bag-of-words models

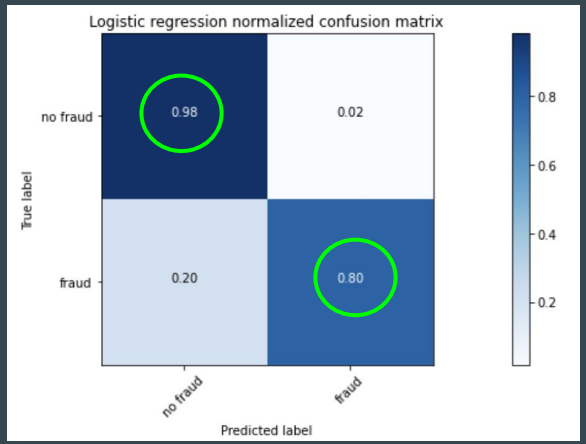
- Bag-of-words models have higher accuracy than decision tree (97+% vs 82%) and higher hit rate (98+%).
- All models show fairly equal performance in identifying fraudulent job postings with the following TN rates: 81% for decision tree, 76% for naive bayes, 80% for logistic regression.
- Given highest TN rate and easier implementation, decision tree can be considered as the best model.



Decision tree, 82.52% accuracy
(model based on primary attributes)



Naive bayes, 97.86% accuracy
(model based on bag-of-words attributes)



Logistic regression, 97.56% accuracy
(model based on bag-of-words attributes)

Model comparison: would it make sense to combine them?

Decision tree and logistic regression are compared based on overlapping segment of test data. Despite small size of the sample, it seems that there is added value in deploying the models in ensemble or sequentially.

If the models were used together, 38% more fraudulent job postings would have been captured

However, such an implementation would bare **increased complexity** and **computational load.**

Therefore, it is recommended to conduct EVF analysis prior to deployment.

18 fraudulent job postings in the shared part of test data:		
Correctly predicted by both models	10	56%
Completely missed by both models	2	11%
Identified by decision tree, missed by log regression	5	27%
Identified by log regression, missed by decision tree	1	6%

Conclusions & business value: what are the learnings?

- The study confirmed H1 and showed that automatic classification of fraudulent job postings **is possible with 82% accuracy or higher.**
- The study confirmed H2 and identified that **salary**, **information about employer** (logo, company profile) and **required education and experience** are the main predictors..

Recommendation for business

- Decision tree is recommended as the most efficient model to identify fraudulent job postings. If decision tree models shows low prediction confidence (eg. <70% or other benchmark defined by business), additional testing with logistic regression (bag-of-words) is recommended..
- If justified by cost-benefit analysis, ensemble deployment of decision tree and logistic regression can be considered.