A Comparative Study on Fake Job Post Prediction Using Different Data mining Techniques

ABSTRACT

In recent years, due to advancement in modern technology and social communication, advertising new job posts has become very common issue in the present world. So, fake job posting prediction task is going to be a great concern for all. Like many other classification tasks, fake job posing prediction leaves a lot of challenges to face. This paper proposed to use different data mining techniques and classification algorithm like KNN, decision tree, support vector machine, naive bayes classifier, random forest classifier, multilayer perceptron and deep neural network to predict a job post if it is real or fraudulent. We have experimented on Employment Scam Aegean Dataset (EMSCAD) containing 18000 samples. Deep neural network as a classifier, performs great for this classification task. We have used three dense layers for this deep neural network classifier. The trained classifier shows approximately 98% classification accuracy (DNN) to predict a fraudulent job post.

**EXISTING SYSTEM**

Many researches occurred to predict if a job post is real or fake. A good number of research works are to check online fraud job advertiser. Vidros [1] et al. identified job scammers as fake online job advertiser. They found statistics about many real and renowned companies and enterprises who produced fake job advertisements or vacancy posts with ill-motive. They experimented on EMSCAD dataset using several classification algorithms like naive bayes classifier, random forest classifier, Zero R, One R etc. Random Forest Classifier showed the best performance on the dataset with 89.5% classification accuracy. They found logistic regression performing very poor on the dataset. One R classifier performed well when they

balanced the dataset and experimented on that. They tried in their work to find out the problems in ORF model (Online Recruitment Fraud) and to solve those problems using various dominant classifiers.

Alghamdi [2] et al. proposed a model to detect fraud exposure in an online recruitment system. They experimented on EMSCAD dataset using machine learning algorithm. They worked on this dataset in three steps- data pre-processing, feature selection and fraud detection using classifier. In the preprocessing step, they removed noise and html tags from the data so that the general text pattern remained preserved. They applied feature selection technique to reduce the number of attributes effectively and efficiently. Support Vector Machine was used for feature selection and ensemble classifier using random forest was used to detect fake job posts from the test data. Random forest classifier seemed a tree structured classifier which worked as ensemble classifier with the help of majority voting technique. This classifier showed 97.4% classification accuracy to detect fake job posts.

Huynh [3] et al. proposed to use different deep neural network models like Text CNN, Bi-GRU-LSTM CNN and Bi- GRU CNN which are pre-trained with text dataset. They worked on classifying IT job dataset. They trained IT job dataset on TextCNN model consisting of convolution layer, pooling layer and fully connected layer. This model trained data through convolution and pooling layers. Then the trained weights were flattened and passed to the fully connected layer. This model used softmax function for classification technique. They also used ensemble classifier (Bi-GRU CNN, Bi-GRULSTM CNN) using majority voting technique to increase classification accuracy. They found 66% classification accuracy using TextCNN and 70% accuracy for Bi-GRU- LSTM CNN individually. This classification task performed best with ensemble classifier having an accuracy of 72.4%.

Zhang [4] et al. proposed an automatic fake detector model to distinguish between true and fake news (including articles, creators, subjects) using text processing. They had used a custom dataset of news or articles posted by PolitiFact website twitter account. This dataset was used to train the proposed GDU diffusive unit model. Receiving input from multiple sources simultaneously, this trained model performed well as an automatic fake detector model.

Disadvantages

1) The system is implemented by Conventional Machine Learning.

2) The system doesn’t implement for analyzing large data sets.

**PROPOSED SYSTEM**

The system has used EMSCAD to detect fake job post. This dataset contains 18000 samples and each row of the data has 18 attributes including the class label. The attributes are job\_id, title, location, department, salary\_range, company\_profile, description, requirements, benefits, telecommunication, has\_company\_logo, has\_questions, employment\_type, required\_experience, required\_education, industry, function, fraudulent (class label). Among these 18 attribute, we have used only 7 attributes which are converted into categorical attribute. T elecommuting, has\_company\_logo, has\_questions, employment\_type, required experience, required\_education and fraudulent are changed into categorical value from text value. For example, “employment\_type” values are replaced like this- 0 for “none”, 1 for ‘full-time”, 2 for “part-time” and 3 for “others”, 4 for “contract’ and 5 for “temporary”. The main goal to convert these attributes into categorical form is to classify fraudulent job advertisements without doing any text processing and natural language processing. In this work, we have used only those categorical attributes.

**Advantages**

1) The proposed has been implemented EMSCAD technique which is very accurate and fast.

2) The system is very effective due to accurate detection of Fake job posts which creates inconsistency for the job seeker to find their preferable jobs causing a huge waste of their time.

**SYSTEM REQUIREMENTS**

➢ **H/W System Configuration:-**

➢ Processor - Pentium –IV

➢ RAM - 4 GB (min)

➢ Hard Disk - 20 GB

➢ Key Board - Standard Windows Keyboard

➢ Mouse - Two or Three Button Mouse

➢ Monitor - SVGA

**SOFTWARE REQUIREMENTS:**

* **Operating system :** Windows 7 Ultimate.
* **Coding Language :** Python.
* **Front-End :** Python.
* **Back-End :** Django-ORM
* **Designing :** Html, css, javascript.
* **Data Base :** MySQL (WAMP Server).