Build a recommendation system that recommends the top 5 relevant writers for the assignment.

Based on historical data, what we have is:
- Writer information
- Assignment information
- Other parameters.
- Each assignment has a writer assigned to it.

Outline:
- Data preprocessing
- Build K-means clustering ML model where each cluster has a similar kind of writers.
- Auto-assign the cluster label to the assignment data points using writer information.
- Build a classification model that helps in recommending a cluster.
- Recommend the top 5 relevant writers from the predicted cluster.

Data Preprocessing on Writer information:
- We need to convert data set features into numerical form.
- Basic approach to convert non-numerical features to numeric features is categorical to numeric form using label encoding, one-hot encoding.
- Textual data to numeric form using bag of words, TFIDF & Word embedding.
- Apply label binarizer on Gender feature that returns 1 for Male and 0 for Female.
- One hot encoding over Language, Genre, Vertical, Availability features, and each of this feature returns sparse data.
- Other features such as Pay, WPD, Average Score are in numerical form only. So, no need to convert but we can apply binning over these numeric data to get some ranges/bins that help in improving the accuracy of the model if required.

Clustering algorithms (K-means algorithm):
- *The objective of building this model is that we can create some clusters where the same kind of writers fall into one cluster and another same kind of writers into another cluster and so on.*
- Using the above-pre-processed data, we can apply K-means algorithm to create clusters for similar kinds of writers.
- How many numbers of clusters to be developed? To choose a value for K/ number of clusters to be chosen, we can opt for the value by inferring from the Genre feature.
- Still, after fitting the K-Means model, if we can't build better clusters then we can apply the Elbow method to get the best value for "number of clusters" and re-fit the model again.
- Once we finalized the clustering model and then we can use it for labeling with assignment data points.

Auto assigns the cluster number/label to the assignments by inferring the writer information, to do this follow the below steps:
- As we know we have historical data, and each assignment has a writer assigned.
- Using this given information, we can choose the relevant cluster for the assignments.
- For example, take an assignment and its writer, find that exact cluster where this writer is present. Once we find the exact cluster, then assign that cluster label to the chosen assignment.

- Do the above steps for all the assignments and create a classification dataset where the feature is the assignment parameters and the target column is the cluster label.
- Next step is to build a classification model that recommends a cluster label.

Build a classification model to predict the writer's cluster:
- Assignment parameters/features will be taken to build the model.
- Just like we have done data pre-processing on the writer parameters, a similar kind of approach we will follow.
- Due date feature we can drop for now. We will use this feature in the final post-processing during recommending users.
- Here, input features are assignment parameters and the target column is our cluster label.
- Experiment with a few machine learning models. As per experience, we can start with a Decision tree / random forest ML algorithm and build a classification model over the data.
- To improve the model or choose the best hyperparameters for the model, apply GridCV search class from scikit-learn to get the best parameters for the model.
- For model evaluation, check the accuracy, F1-score, and other metrics as well to better understanding.
- This model will predict the cluster label for any assignment.
- Once, we predict a cluster label(same kind of writers), then we have to choose the top 5 relevant users for the assignment using further steps.

Choose the top 5 writers in the order of relevancy:

- Above classification model helps to get a cluster label.
- Pick all the writers from the predicted cluster.
- Build a similarity measure algorithm using content-based filtering by considering other factors such as due date, score, status, WPD.
- Sort the user based on relevancy score and chose the top 5 writers.

Further improvements:
- Instead of a clustering-based model, we can build a classification model by putting some manual efforts to improve the accuracy for identifying the cluster label to a great extent.
- Use the neural network approach for classifying the cluster label.
- Nowadays, Deep learning algorithms are also very useful to build a recommendation system if we have a good amount of data.
- Heavy post-processing and different measures can be used to reduce the error.
- Genre, Words, availability, score, and status parameters can be used for relevancy score.