## Team Members

**SRI VISHNU. M :** [**srivishnu1103@gmail.com**](mailto:srivishnu1103@gmail.com)  **(Team Leader)**

**SATHIS KUMAR. G :** [**g.sathiskumar22@gmail.com**](mailto:g.sathiskumar22@gmail.com)

**SANJAY KUMAR. R :** [**rsanjaykumar1319@gmail.com**](mailto:rsanjaykumar1319@gmail.com)

**MOHAMMED NOORULLA. S :**

**noorullasnoorul123@gmail.com**

Fake News Detection using NLP

In fake news detection using Natural Language Processing (NLP), loading and preprocessing the dataset are essential steps. Here's a high-level overview of the process:

1. Loading the Dataset:

Obtain a labeled dataset that contains news articles or text data with labels indicating whether they are fake or real news.

Common sources for such datasets include Kaggle, research repositories, or custom data collection efforts.

2. Data Cleaning:

Remove any irrelevant or extraneous information from the text, such as HTML tags, special characters, or numbers.

Tokenize the text into words or subwords, depending on your NLP model.

3. Text Preprocessing:

Convert text to lowercase to ensure uniformity.

Remove stop words (common words like "the," "and," "in") to reduce noise.

Perform stemming or lemmatization to reduce words to their root form.

4. Data Splitting:

Split the dataset into training, validation, and test sets. A common split might be 70% for training, 15% for validation, and 15% for testing.

5. Padding:

If you're using neural networks, pad or truncate the sequences to a fixed length to ensure consistent input sizes.

6. Data Loading:

Use data loaders or generators to efficiently load and feed the data to your NLP model in batches. Libraries like PyTorch and TensorFlow provide tools for this.

7. Label Encoding:

Encode the labels (fake or real) as numerical values, typically 0 and 1.

8. Balancing the Dataset (Optional):

Check for class imbalance and consider techniques like oversampling or undersampling to balance the dataset.

9. Data Augmentation (Optional):

For text data, data augmentation techniques are less common than in image processing, but you can consider methods like synonym replacement or adding noise to text to increase dataset variety.

Once you've completed these preprocessing steps, you can use the preprocessed data to train and evaluate your fake news detection NLP model, which could be based on various algorithms such as recurrent neural networks (RNNs), convolutional neural networks (CNNs), or transformer-based models like BERT.

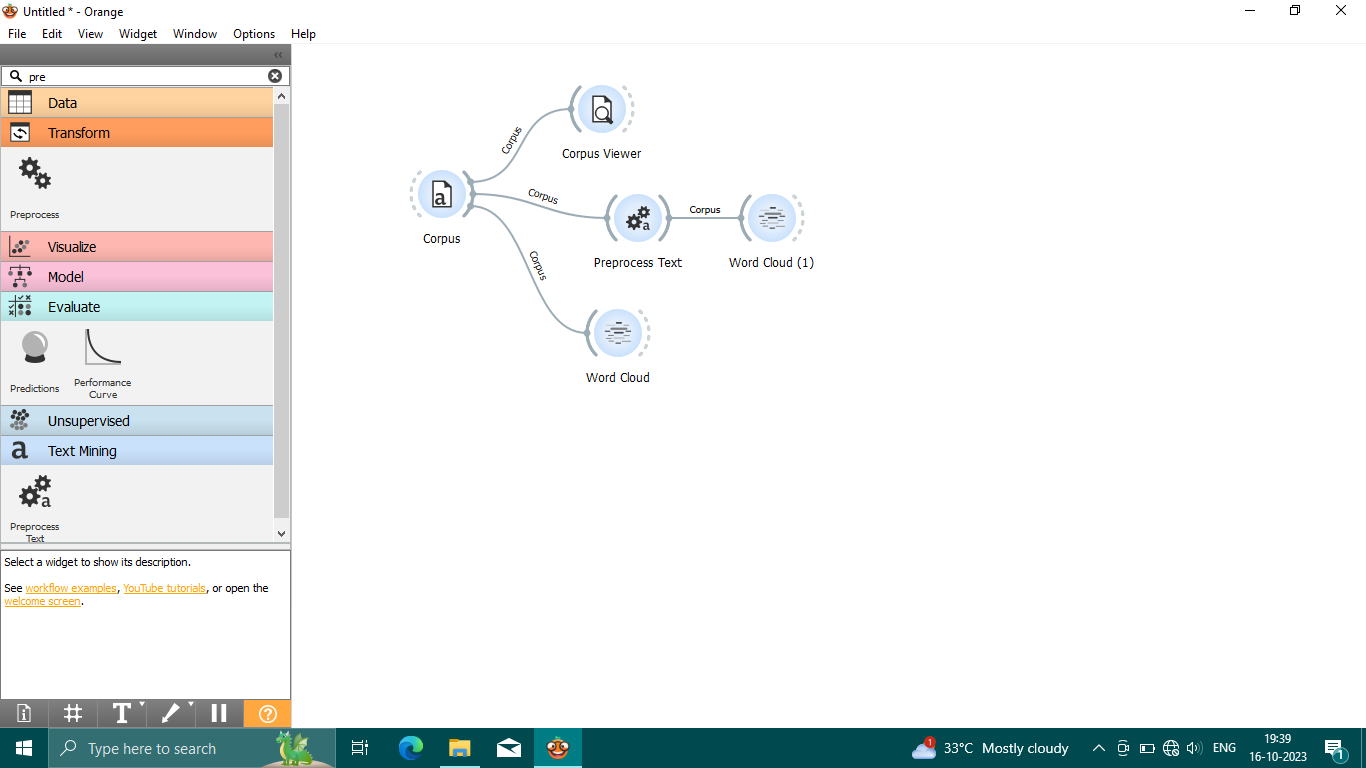
Remember that the specific preprocessing steps may vary depending on the characteristics of your dataset and the NLP model you're using.

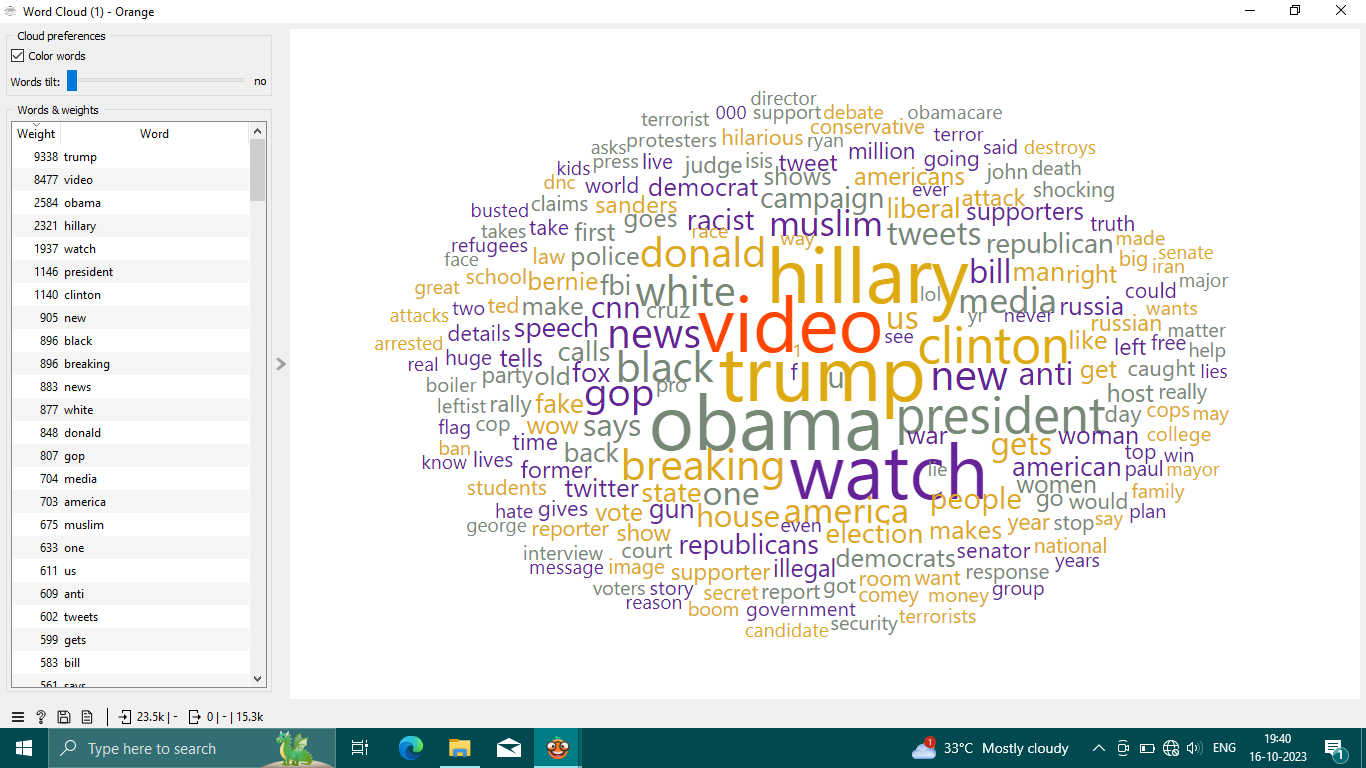
For loading and preprocessing here we are using orange software or tool.

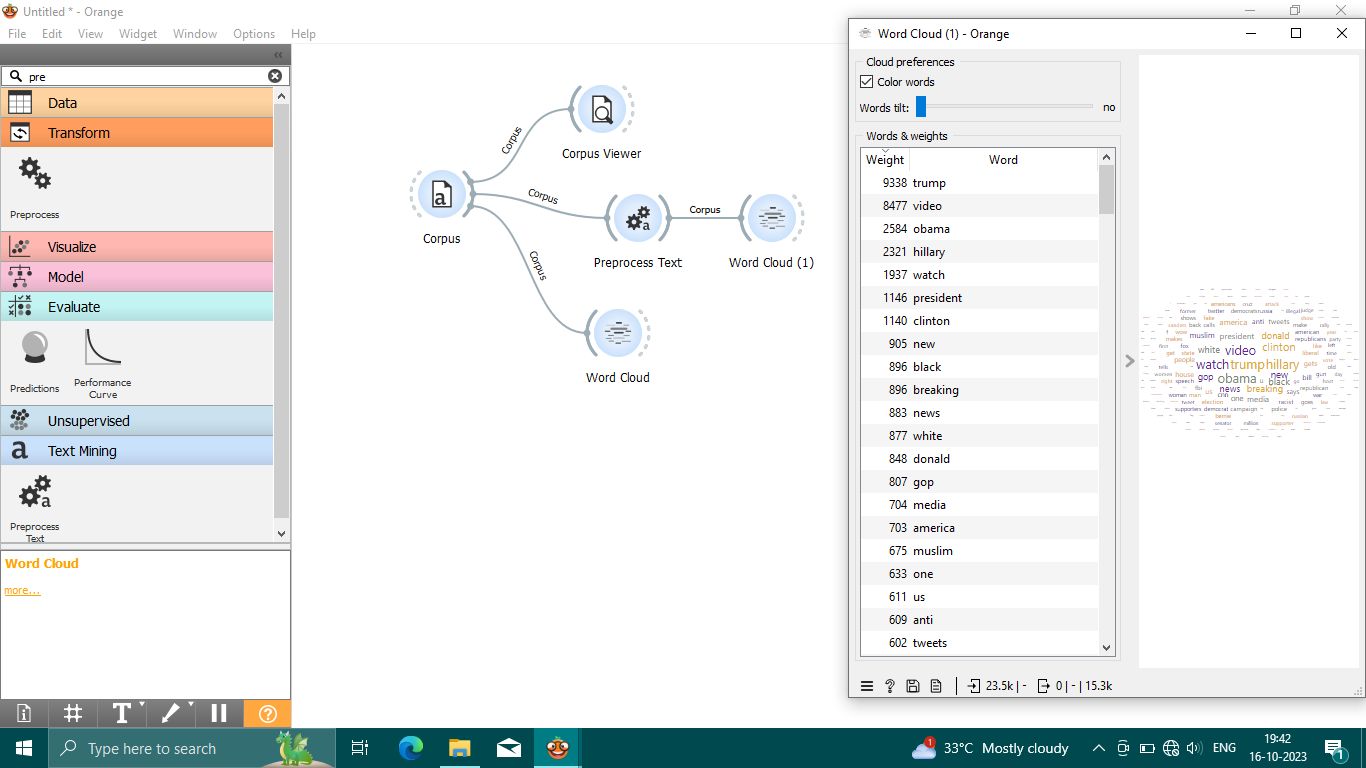
The following images represents the loading and preprocessing of the kaggle data set using orange tool.

Loading and Preprocessing using orange software:

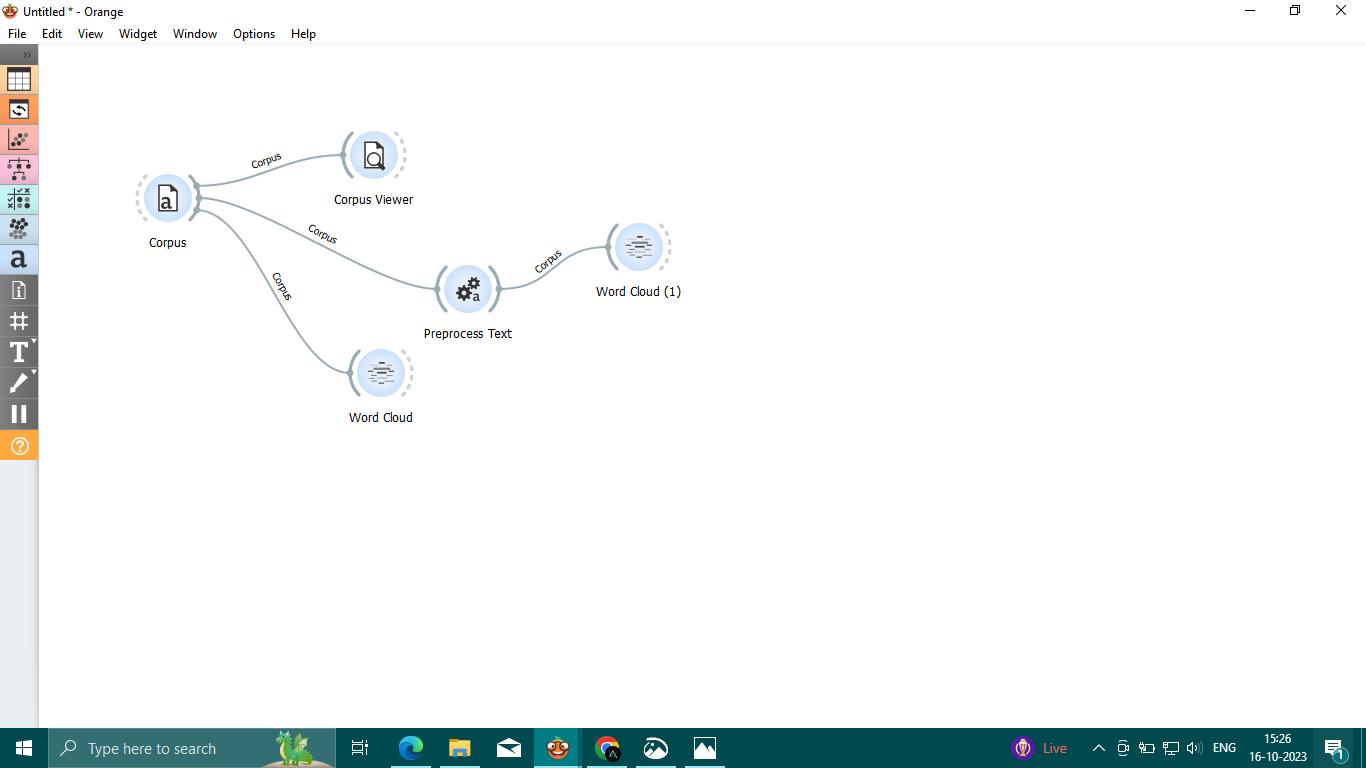
For fake datasets.

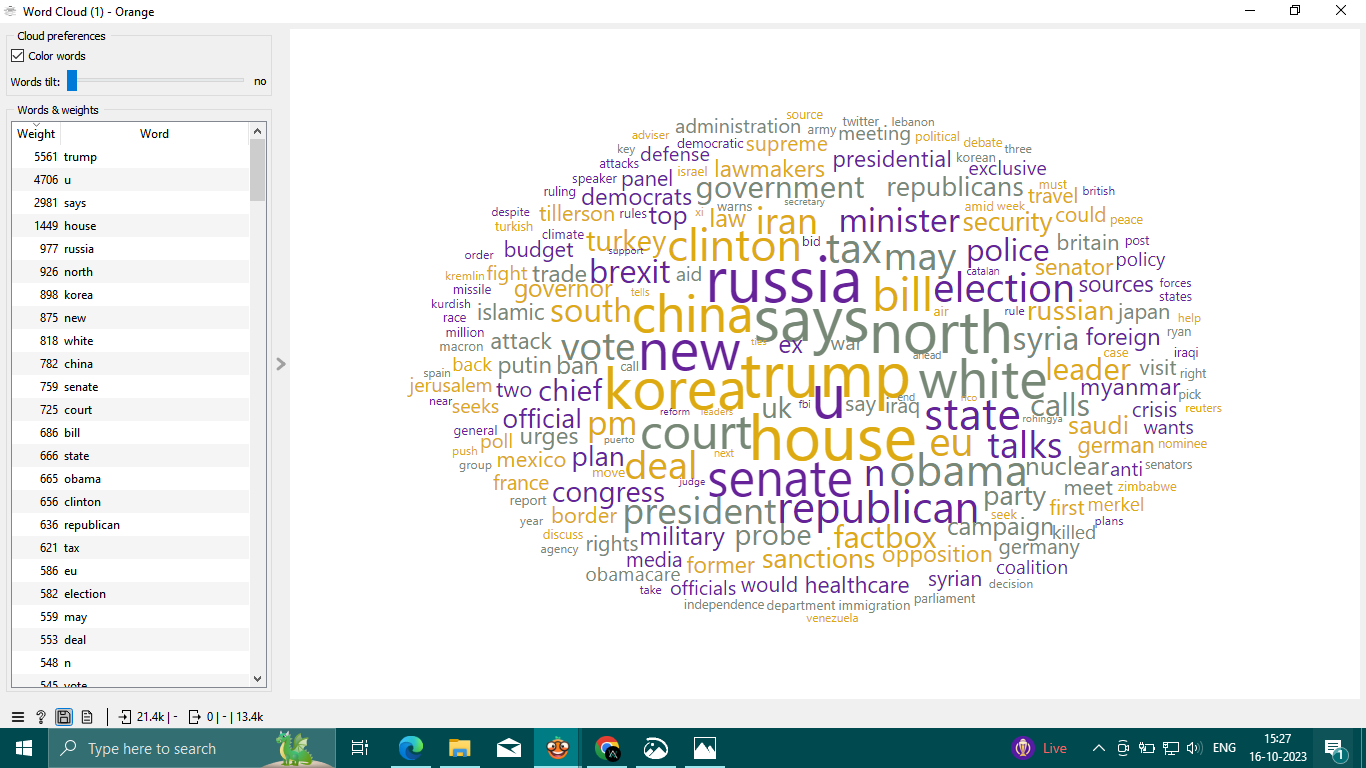


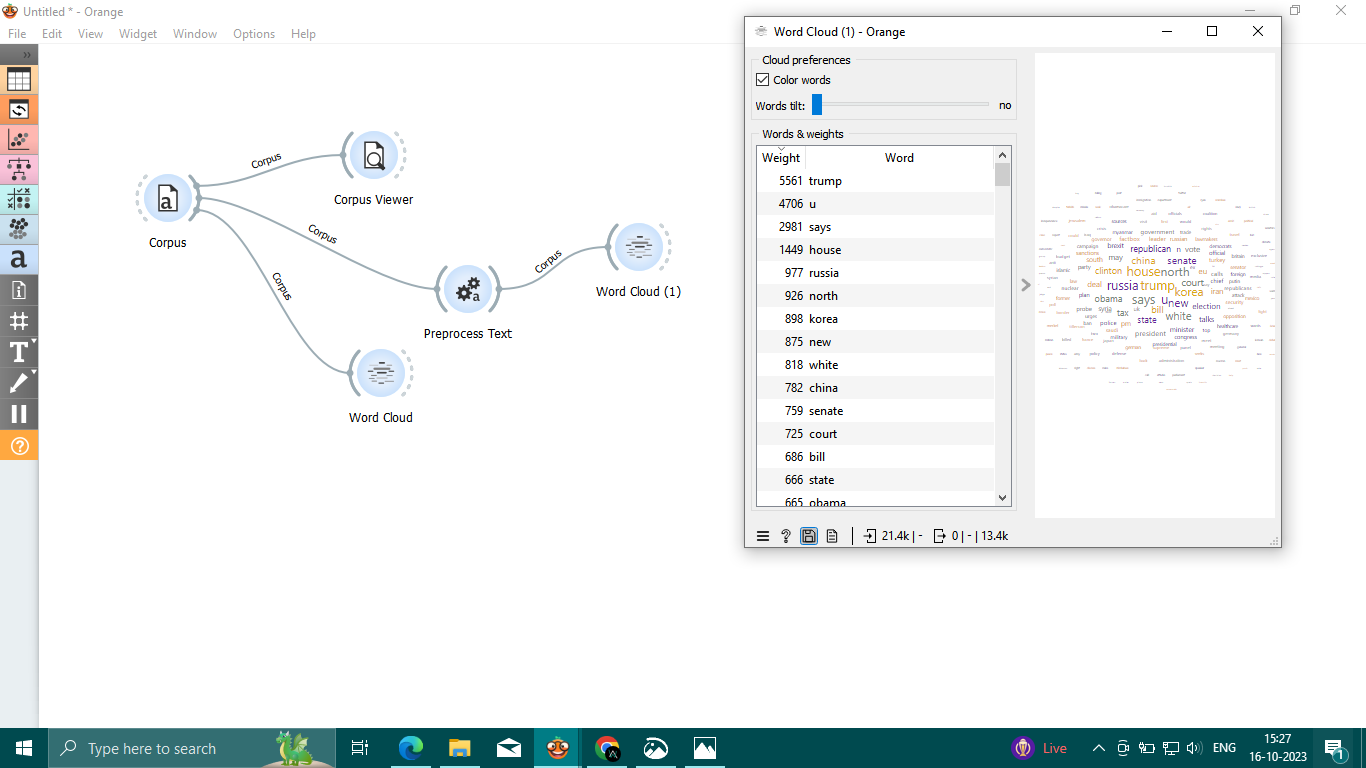




For true datasets.







Conclusion:

Thus the loading and preprocessing of the kaggle data set (fake news detection using NLP) has been successfully processed by the using orange tool.