Machine Learning Coursework Report

By

Student Registration Number: 210502451

In partial fulfilment of the requirements for the module:
ST3189: Machine Learning

UNIVERSITY OF LONDON

CONTENTS

INTRODUCTION

This report delves into the application of machine learning techniques on real-world datasets to uncover patterns, make predictions, and classify information. Focused on Unsupervised Learning, Regression, and Classification. The analysis aims to address specific research questions and present findings in an accessible manner. Leveraging datasets from UCI, we will explore the intricacies of these techniques, contributing to a broader understanding of their application in practical scenarios.

1. REGRESSION MODEL

A regression model in machine learning predicts a numerical outcome (dependent variable) based on one or more input features (independent variables).

A. Dataset Information

The Abalone dataset, contributed to UCI in 1995 by marine biologist Warwick J Nash, aims to predict the age of abalones based on physical characteristics. Abalones are marine mollusks with a spiral shell, and determining their age is crucial for fisheries management. The dataset includes attributes like sex, length, diameter, and weights. The primary task is regression, predicting the age (in rings) of abalones based on these features.

B. Features

Features	Length
Sex	Whole_weight
Diameter	Shucked_weight
Height	missing_weight*
Viscera_weight	Shell_weight

Table 1: Features

*The missing_weight feature was generated from the difference between Whole_weight and Viscera, Shell & Shucked weights.

C. Analysis

The Whole weight feature is derived as the cumulative sum of Viscera, Shell, and Shucked weight features. However, discrepancies may arise due to liquid loss after separating these parts for measurement. Consequently, the missing_weight feature was introduced to account for such loss.

A regression plot of each feature against our target variable, Rings, reveals their correlation. Figure 1 demonstrates a positive correlation between all features and the target variable, albeit some features exhibit stronger linear correlation than others.

Figure 2 highlights a significant issue of multicollinearity, indicating strong correlations among the weights and measurements of the abalone. Notably, the P-value of our Length feature is 0.839, suggesting it lacks statistical significance. Consequently, we have opted to exclude it from our feature set.

```
Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The smallest eigenvalue is 9.62e-29. This might indicate that there are
strong multicollinearity problems or that the design matrix is singular.

P-values for each feature in OLS Linear Model:
const             1.525201e-02
Length            8.387534e-01
Diameter          6.698444e-07
Height            4.519697e-12
Whole_weight      1.525201e-02
Shucked_weight    1.525201e-02
Viscera_weight    1.525201e-02
Shell_weight      1.525201e-02
missing_weight    1.525201e-02
Sex_M             1.525201e-02
Sex_F             1.525201e-02
Sex_I             1.525201e-02
```

Figure 2: P-Values of Features

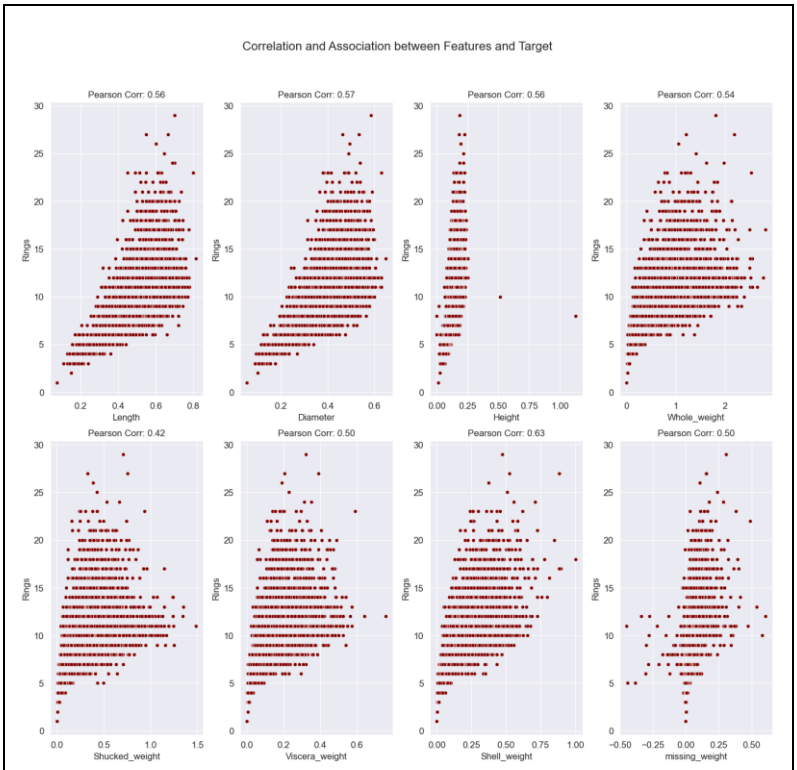

Figure 1: Regression Plots of Features vs Rings

D. Research Questions

I. How would removing outliers affect the model?

It is essential to establish a reference model to assess both models containing outliers and those without outliers. Various algorithms are at our disposal for model creation, and we opt for employing the Ridge Regression model. Ridge regression proves to be adept at managing multicollinearity, addressing scenarios where independent variables exhibit correlation. In the absence of regularization, multicollinearity can result in precarious and untrustworthy estimates of regression coefficients, a challenge that Ridge regression effectively mitigates. Moreover, the algorithm exhibits a diminished sensitivity to outliers when compared to ordinary least squares. The regularization term plays a crucial role in diminishing the impact of extreme observations, enhancing the overall robustness of the model.

Outliers, characterized as data points markedly divergent from the rest of the dataset, represent abnormal observations capable of distorting the data distribution. Detecting and removing outliers is crucial for ensuring that the trained model effectively generalizes across the valid range of test inputs.

However, one thing to note is that in some cases, the outliers may contain meaningful information about the underlying patterns in the data. Removing them might eliminate crucial insights that the model needs for accurate predictions.

A K-Fold Cross Validation was performed on both models to see how well the model generalizes over different sets of unforeseen data. Table 2 below shows the average scores with and without outliers. We can see that there seems to be a contradictory scenario as both the R-Square and Root Mean Square Error (RMSE) decrease after removing the outliers.

Removing outliers might have helped the model capture the true underlying relationships better, despite slightly sacrificing fit on individual training points.

Model	R Square	Root Mean Square Error
With Outlier	0.514	2.237
Without Outliers	0.496	2.066

Table 2:Linear Model Scores

II. Does the data fit a linear model?

For the dataset to be a good fit for a linear model, there are four assumptions that are made.

1. Linearity: The relationship between Independent Variables and Dependent Variables should be linear
2. Homoscedasticity: The variance of residuals should be constant

3. Independence: The independent variables should have little to no correlation to one another
4. Normality: The residuals follow a normal Distribution

From our previous observations from visualization and performing statistical tests, the data has failed two assumptions, Linearity, and Independence. Figure 3 below, shows the density plot of our residuals.

While the distribution of the residual terms shows a slight skew, it is close to normal given the number of data points we collected. This aligns with the central limit theorem, which suggests that larger sample sizes naturally tend towards a normal distribution. It is important to remember that perfect, bell-shaped curves are rare in real-world data, and minor deviations should not necessarily raise concerns.

However, the assumption of homoscedasticity plays a critical role in linear regression. Violations of this assumption can lead to biased standard errors, which are pivotal for significance tests and confidence interval calculations. To assess homoscedasticity, one can examine a residual vs. fitted values plot, where the presence of a cone-shaped scatter plot indicates heteroscedasticity.

Figure 4 below is a scatter plot of the residual vs the predicted values. We can see a cone like shape present indicating that the assumption of homoscedasticity is violated.

As such, the dataset might not be a good fit for a linear model.

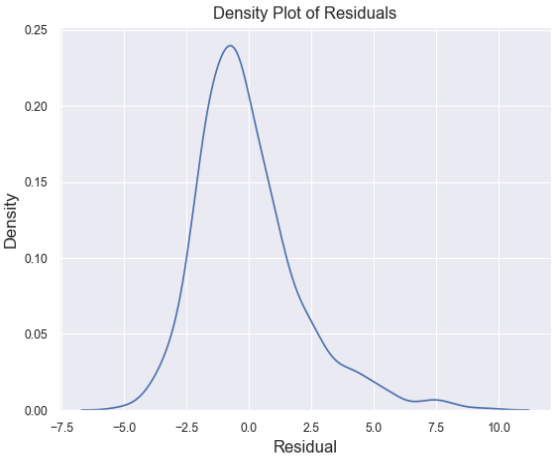

Figure 3: Density Plot of Residuals

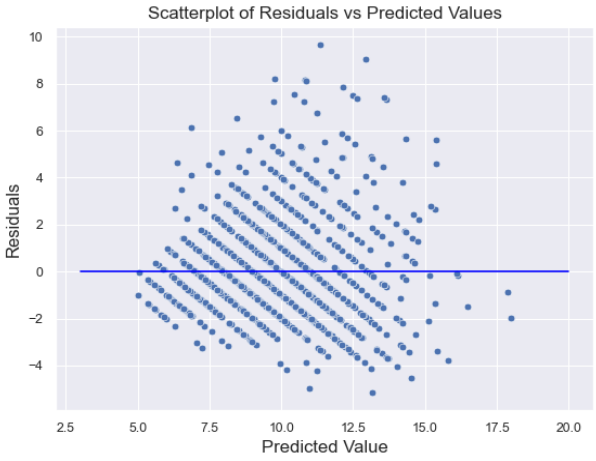

Figure 4: Scatter Plot

III. *Would a non-linear model make a better model?*

After conducting visual and statistical analyses on our linear regression models, we uncovered compelling evidence suggesting a non-linear relationship between the features and the number of rings.

Acknowledging this non-linearity, we opted for a boosting tree algorithm as our predictive model. This algorithm is well-suited for handling non-linear problems by amalgamating the predictions of multiple sequentially built decision trees, with each tree focusing on addressing the shortcomings of the previous one.

To implement this approach, we utilized the CatBoost library. In hyperparameter tuning, we adjusted parameters such as the number of trees, learning rate, bootstrap methods, and l2_leaf_reg to achieve optimal performance. The learning rate plays a crucial role in determining the number of iterations required for the model to learn; a higher learning rate leads to fewer iterations needed to learn, and vice versa. Additionally, we introduced the bootstrap methods and l2_leaf_reg parameters to further fine-tune our model's performance. Figure 5 below illustrates the optimal learning rate and iterations required to minimize our RMSE score.

Using the optimal parameters shown in figure 5, we have managed to lower our RMSE to 1.87. Indicating that a Non-Linear Model is a better suited for this dataset.

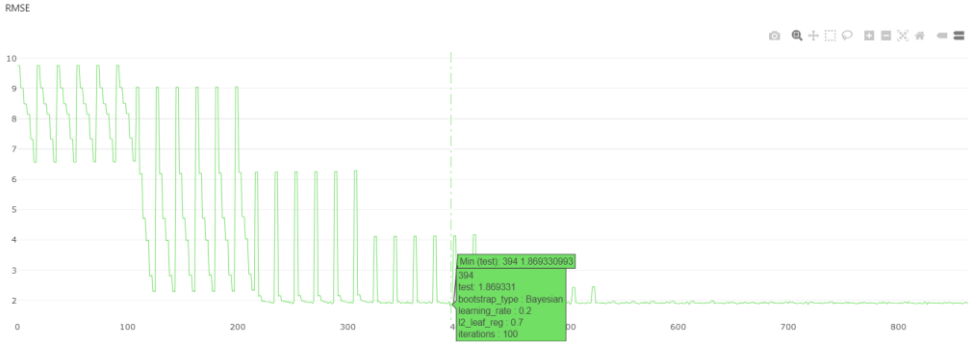

Figure 5: Hyper-Parameter Tuning

2. CLASSIFICATION MODEL

Classification is a supervised machine learning technique aimed at predicting the correct label for a given input data..

A. *Dataset Information*

The Adult dataset, also known as UCI Adult Census Income or simply "adult", is a widely used benchmark dataset in machine learning tasks like classification and prediction. It originated from the 1994 U.S. Census data and contains anonymized demographic information along with an individual's annual income label.

B. *Features*

Numerical Features	Description	Categorical Features	Description
Age	the age of an individual	Relationship	represents what this individual is relative to others.
Fnlwgt	final weight. This is the number of people the census believes the entry represents.	Race	Descriptions of an individual's race
Educationnum	the highest level of education achieved in numerical form.	Sex	the sex of the individual
Capitalgain	capital gains for an individual	Education	the highest level of education achieved by an individual.
Capitalloss	capital loss for an individual	maritalstatus	marital status of an individual
hoursperweek	the hours an individual has reported to work per week	Occupation	the general type of occupation of an individual
-	-	Nativecountry	country of origin for an individual
-	-	Workclass	a general term to represent the employment status of an individual

Table 3:Adult Census Features

C. Analysis

Firstly, we chose to remove the education feature as it was just a categorical version of Educationnum. Secondly, we drop 'Fnlwgt' as this feature was created while studying the dataset and was not acquired during the survey. Next, we plot a correlation heatmap of our numerical features and a Variance inflation factor (VIF) analysis on our feature set to identify any multi-collinearity effect. Figure 6 below shows that our numerical features are not significantly correlated to one another. Additionally, we can view that 'fnlgwt' has a correlation close to zero indicating the feature is not significant feature. As such we have dropped the feature.

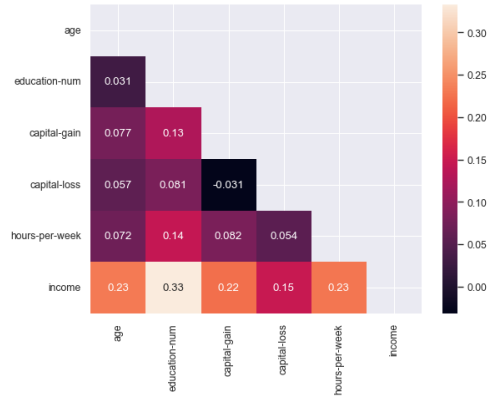

Figure 6: Heatmap of Numerical Features

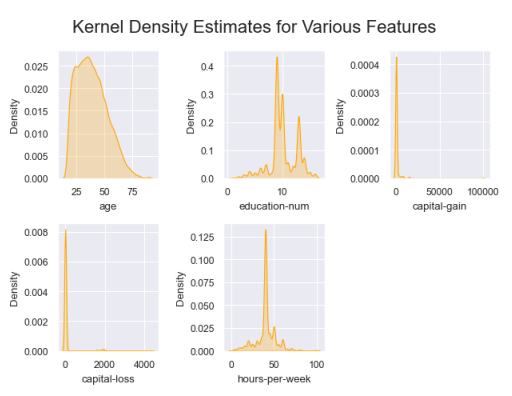

Figure 7: Density Plot of Numerical Features

Multicollinearity is considered significant when the Variance Inflation Factor (VIF) exceeds a threshold of 5 to 10 (Jong Hae Kim, 2019). In our analysis, it is evident that six features have VIF scores exceeding 5.

However, performing only one statistical test is not a clear indication of multicollinearity.

Numerical Feature	VIF Score	Categorical Feature	VIF Score
age	8.789	Race	17.882
Capitalgain	1.045	Sex	4.422
Capitalloss	1.062	maritalstatus	4.056
Educationnum	15.743	Occupation	3.385
hoursperweek	11.827	Relationship	2.641
		Nativecountry	28.383
		Workclass	8.413

Table 4: Adult Census VIF Scores

Figure 7 to the left allows us to view the distribution of our numerical columns. We can view that both capital gain and loss are rightly skewed. Whereas the others are close to a normal distribution. These are important considerations to consider when we are pre-processing our data to be fed into the model.

A dataset with imbalanced class proportions is referred to as imbalanced. Majority classes are those that constitute a significant portion of the dataset, while minority classes make up a smaller proportion (Google ,2023).

Figure 8 below is a bar plot visualizing the proportion of our classes. We can view that there is a mild class imbalance of ~75% for our majority class and ~25% for our minority class.

Figure 8: Class imbalance

D. Research Questions

I. How does Resampling Techniques affect a imbalanced Dataset

To begin with a simple algorithm selection for model creation, the choice leans towards utilizing Support Vector Machine (SVM). SVM constructs a hyperplane to optimally separate two classes while maximizing the margins. Nevertheless, when applied to imbalanced datasets, SVM models may exhibit a bias towards the majority class in the placement of the separating hyperplane. This imbalance can detrimentally affect the model's performance, particularly with regards to the minority class (Batuwita, R & Palade, V. ,2012). As such, we would like to see how the different techniques such as resampling and performing cost sensitive training will affect the model.

In oversampling, exemplified by Synthetic Minority Oversampling Technique (SMOTE), the number of samples in the minority class is increased to match the majority class. SMOTE addresses imbalanced datasets by generating synthetic samples for the minority class, interpolating new data points between existing minority class samples.

Conversely, undersampling methods like NearMiss reduce the number of samples in the majority class to match the minority class. NearMiss achieves a balanced class distribution by strategically eliminating examples from the majority class, focusing on instances where two different classes are in proximity and selectively removing majority class instances to enhance class separation.

These resampling techniques are vital for mitigating the impact of class imbalance on model performance, thereby improving prediction accuracy and fairness across all classes.
The table below represents the F1 scores of the different techniques.

Baseline	SMOTE	NearMiss
0.807	0.831	0.760

Table 5: Oversampling vs Undersampling

Performing SMOTE has increased our scores by 3%, proving that performing resampling techniques does increase accuracy for this dataset. However, NearMiss might have removed an excessive number of data points, resulting in information loss and a suboptimal model performance.

This nuanced combination of oversampling and Undersampling techniques holds relevance for Support Vector Machines (SVM). The strategic application of these methods may contribute to the creation of a more optimal hyperplane, facilitating the effective separation of the two classes in SVM classification.

The table below represents the F1 Scores of the different techniques.

SMOTEENN	SMOTEK
0.950	0.883

Table 6: Combined Undersampling and Oversampling Techniques

In Table 7 above, it is evident that SMOTEEN has significantly improved our scores compared to SMOTE. Specifically, SMOTEENN has elevated the score from 0.807 to 0.950, indicating a substantial enhancement.

We can scale our data and then perform Principal Component Analysis (PCA) to reduce dimensionality for interpretation and visualization purposes.

Figure 10 below presents a 3-dimensional visualization of our features, where the target label serves as the hue. By employing SMOTEENN, we gain insights into why SVM found it more feasible to establish a hyperplane between the two classes.

Figure 10: Visualizing data of Baseline vs SMOTEENN

II. How would Pruning a Decision Tree affect the model

When dealing with imbalanced data, random forest or boosting algorithmic models works well with them. However, we would like to view the performance increase/decrease when we prune our decision tree.

Decision trees, unlike most machine learning models, often thrive without numerical feature scaling. That's because decision trees make decisions based on whether something is above or below a certain number, rather than how big or small it is compared to other numbers. So, even if we change the size of our numbers, it doesn't change how the decision tree makes its choices. Plus, decision trees pick which features are most important by looking at how much they help to separate our data into different groups, not by how big or small the numbers are.

However, scaling is not always irrelevant. It can benefit cases with vastly different feature scales or algorithms using distance metrics. From our observations, we have tested the models to confirm if feature scaling would provide a better or less accurate model.

The evaluation metrics employed in these tests focus on the Weighted F1-Score. This metric calculates the harmonic mean of precision (accuracy of positive predictions) and recall (ability to correctly identify true positives) for each class independently. When averaging these scores, a weighted approach is used, considering the number of true instances for each class. This approach addresses class imbalance, providing a more realistic metric for model performance.

The results are shown in the table below.

w/Feature Scaling	wo/Feature Scaling
0.818	0.821

Table 7: Comparing Feature Selection

We can observe that for this dataset, it would be better to not perform feature scaling on the account that we reduce the need of unnecessary processing and to increase our score.

Pruning, a method linked with decision trees, involves diminishing the size of the trees by eliminating segments that do not contribute significantly to classifying instances. Among various machine learning algorithms, decision trees are particularly prone to overfitting, and proficient pruning can mitigate this risk effectively. The greater the alpha value, the more trees are pruned. As such we must pick an alpha that is not too low which leads to overfitting and not too high which leads to having too many trees removed.

Figure 7 below provides a visual representation of the impact of increasing alpha on the accuracy of our model. When alpha is set to zero, with the default parameters of the DecisionTreeClassifier maintained, the tree tends to overfit the training data, resulting in 100% training accuracy and 81% testing accuracy. However, as the alpha parameter increases, more of the tree is pruned, leading to the development of a decision tree that generalizes more effectively. In this scenario, setting alpha to approximately 0.000145 maximizes the testing score to 82.5%.

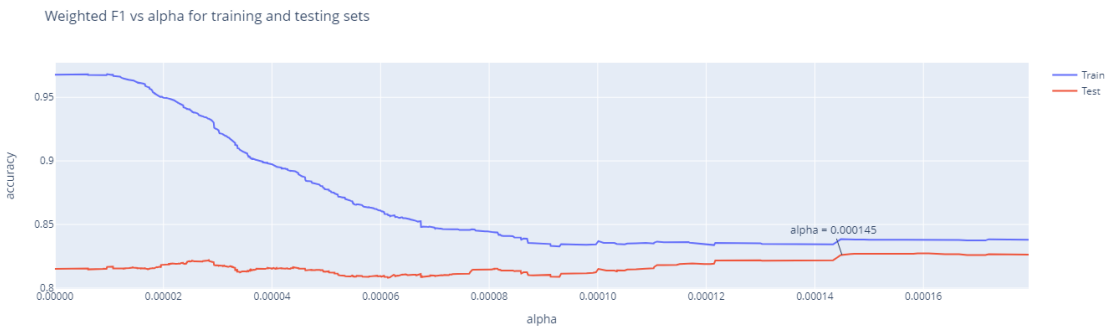

Figure 7: Pruning Tree Score

3. UNSUPERVISED LEARNING

Unsupervised learning is a machine learning paradigm where algorithms are presented with data lacking predefined labels or outputs. Instead, they are tasked with autonomously identifying patterns, structures, or relationships within the dataset.

A. *Dataset Information*

The dataset was obtained from Kaggle containing over 1 million news headlines over a period of 19 years from Australian news Source ABC (Australian Broadcasting Corporation)

B. *Features*

publish_date	headline_text

C. *Analysis*

By analyzing our dataset and applying preprocessing steps such as removing stopwords and tokenization, we were able to generate visualizations to illustrate our findings. Removing stopwords involves eliminating frequently used words such as "the" "is" and "and" etc., which do not contribute significantly to the analysis. Tokenization breaks down the text into individual words or tokens, creating a structured representation of the data for further analysis.

Figure 8 presents the total number of articles per year spanning 19 years. Notably, there is a decline in articles post-2015, indicating a potential shift in publishing trends. On the right, we highlight the most frequently used words across these years, which were extracted after removing stopwords.

Figure 8: Overall text analysis

Examining articles containing disaster-related words, Figure 8 also depicts a significant spike in 2020, likely attributed to the COVID-19 pandemic. Notably, keywords such as "COVID" and "coronavirus" contribute to this increase. These keywords were identified through tokenization and subsequent analysis.

Figure 9 showcases the top 3 words per year, derived from tokenized text after removing stopwords. Until 2020 and 2021, the most used words remain largely consistent. Notably, "COVID," "coronavirus," and "19" dominate these years, with "19" likely referencing COVID-19. Additionally, the term "new" may relate to articles about new COVID strains or cases

Figure 10 illustrates the density and distribution of positive, negative, and neutral sentiment

scores per day. Most headlines exhibit a high neutral sentiment percentage, occasionally accompanied by negative sentiments, reflecting the nature of news reporting. Notably, the tight distribution around zero for positive sentiments reinforces this observation.

Figure 9: Top 3 words used per Year

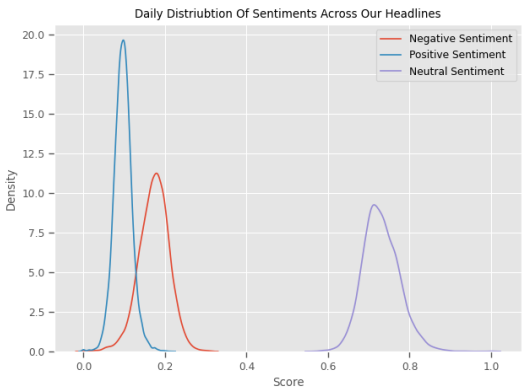

Figure 10: Density plot of sentiments

Analyzing the average sentiment progression over time, sentiments appear relatively stable. However, there are brief periods where negative and positive sentiments fluctuate. For instance, 2014 exhibits slightly more positive sentiments compared to subsequent years, while 2021 headlines predominantly reflect negative sentiments which could have caused by the pandemic. These sentiments were determined through sentiment analysis performed on the tokenized and processed text data.

Figure 11: Daily and yearly Average Sentiments

D. *Research Question*

I. *What are the main topics covered by ABC News over the past 19 years?*

To unveil meaningful topics from our text data, we need to adapt our preprocessing approach. Topic modeling, a potent technique for uncovering latent themes, heavily relies on accurate data preparation. Unlike many other tasks, it prioritizes interpretable topics, necessitating meticulous preprocessing efforts. This scrutiny is exemplified in our analysis of a "Million News Headlines" dataset, emphasizing the critical role of precise data preprocessing.

Precise preprocessing involves filtering out noise, such as stop words, punctuation, numbers, and single-letter words, which contribute little thematic value. Additionally, lemmatization ensures uniform word representation, preventing ambiguity (e.g., transforming "running" to "run"). Focusing on specific part-of-speech (POS) tags—NOUN, ADJ, VERB, ADV—deliberately enhances the relevance and informativeness of the selected words.

In this research question, our choice to concentrate on these specific POS tags was strategic. Nouns capture essential entities and concepts, while adjectives provide descriptive details that enrich topic comprehension. Verbs denote actions or events, shedding light on the dynamics of news coverage, while adverbs add nuance or modify the meaning of verbs, enriching the contextual understanding of headline topics.

By filtering our text data based on these specific POS tags, we ensure that the words chosen for topic modeling are not only pertinent but also rich in information, effectively capturing the main themes and topics discussed in the headlines.

Additionally, employing bigrams and trigrams allows us to capture frequently occurring related concepts, such as "climate change." However, setting a minimum occurrence threshold is crucial to exclude infrequent or insignificant phrases that could distort the data. We opted for an arbitrary threshold of 10 occurrences. This refined dataset was then utilized for topic modeling.

Figure 12 showcases the resulting dashboard, empowering users to explore topics and their associated words, facilitating insightful clustering and analysis.

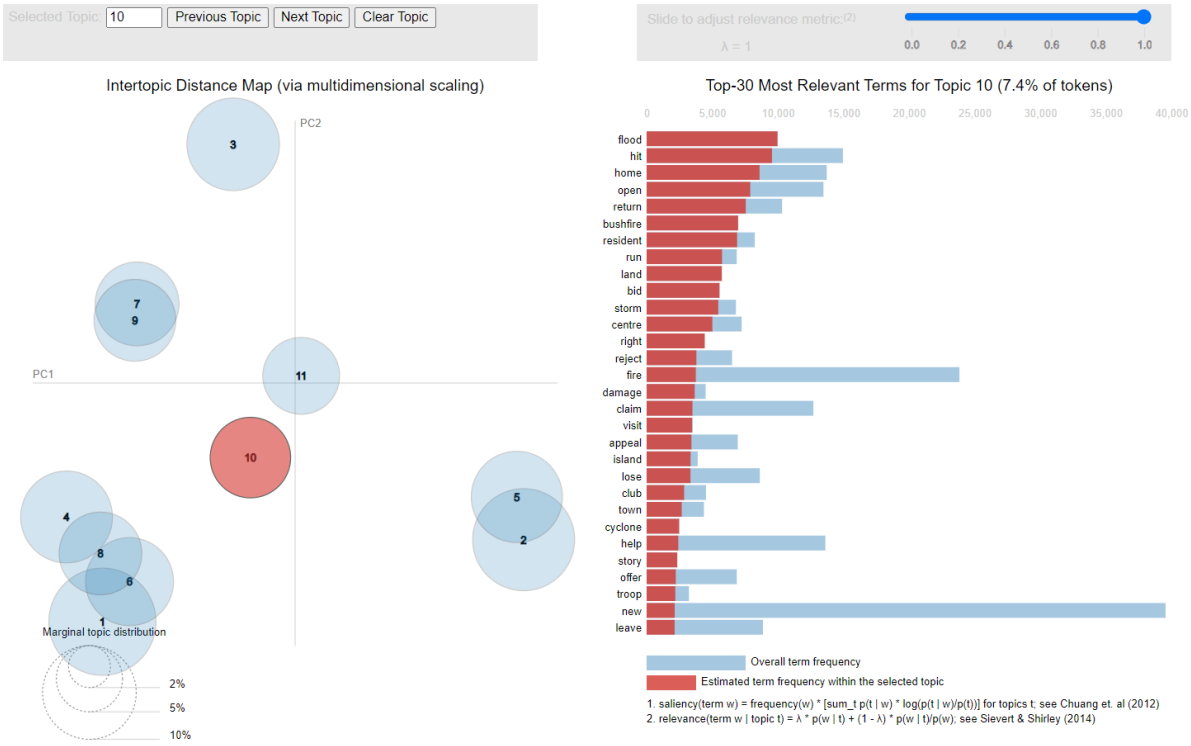

Figure 12: Topic Modelling Results

II. *Can we leverage on Large Language Model (LLM) models to interpret the topics?*

The potential of utilizing LLMs for topic representation holds significant promise for several reasons. Firstly, LLMs automate a process that traditionally demands substantial time and resources: topic interpretation. This process involves assigning high-level labels or phrases to groups of topic words, summarizing, and interpreting their meaning. These labels distil the essence of each topic, making it more understandable and interpretable for humans. With LLMs, the task of assigning meaningful labels to many topics becomes notably faster and more scalable.

Secondly, LLMs are trained on extensive datasets, granting them access to a vast array of information. This enables them to consider diverse perspectives not readily accessible to a single human expert, fostering a more comprehensive understanding of underlying themes within text data.

An exemplary demonstration of LLMs' capabilities in topic classification is evident in the analysis of the "A Million News Headlines" dataset using the LLM Flan T5 by Google. However, the effectiveness of LLM-based topic classification can be further enhanced through prompt engineering. By crafting precise prompts tailored to the specific task and dataset, researchers guide LLMs to generate more accurate and relevant outputs.

Prompt engineering involves designing prompts that effectively steer the LLM toward the desired outcome, including providing context, specifying the desired task, and incorporating relevant keywords or phrases. Through meticulous

prompt engineering, researchers can influence the LLM's understanding of the topic and improve the quality of its output.

As depicted in Table 8 below, the model effectively classified several topics, accurately assigning labels such as "Market Share," "Police: Man charged with murder," and "World Tour." This showcases the LLM's proficiency in identifying core themes within headlines and generating corresponding labels.

However, it's essential to acknowledge the limitations inherent in LLM-based topic classification. As highlighted in the table, the model encountered challenges with certain topics, occasionally assigning inaccurate labels like "gold" "Interview" etc. to topics that were more appropriately categorized differently. This underscores the need for careful consideration and critical evaluation when leveraging LLMs for topic interpretation.

Topic No	Keywords	Label
1	'plan', 'health', 'call', 'water', 'miss', 'urge', 'service', 'council', 'fund', 'rural'	Plan of Action
2	'police', 'man', 'charge', 'court', 'murder', 'jail', 'woman', 'accuse', 'child', 'drug'	Police: Man charged with murder
3	'win', 'final', 'world', 'hour', 'season', 'set', 'weather', 'country', 'beat', 'tour'	World Tour
4	'market', 'rise', 'price', 'power', 'rate', 'business', 'share', 'high', 'sale', 'federal'	Market Shares
5	'fire', 'kill', 'crash', 'die', 'death', 'car', 'attack', 'police', 'dead', 'man'	Fire and car crash
6	'worker', 'job', 'ban', 'work', 'law', 'protest', 'pay', 'tell', 'farm', 'union'	Workers
7	'covid', 'day', 'election', 'case', 'test', 'make', 'coronavirus', 'new', 'go', 'first'	Coronavirus
8	'interview', 'say', 'farmer', 'change', 'industry', 'meet', 'leader', 'drought', 'climate', 'long'	Interview
9	'year', 'new', 'school', 'end', 'time', 'speak', 'action', 'launch', 'old', 'join'	New School
10	'flood', 'hit', 'home', 'open', 'return', 'bushfire', 'resident', 'run', 'land', 'bid'	Flooding
11	'gold', 'show', 'coast', 'driver', 'find', 'beach', 'rail', 'play', 'line', 'safety'	gold

Table 8: LLM Model's Auto Labelling

E. Further Research

The process of interpreting topics following topic modelling can yield varied results depending on the algorithms used to derive keywords and the Large Language Model (LLM) employed for refining topic representation. One well-documented approach worth exploring involves leveraging BERTopic and Llama-2.

BERTopic follows a straightforward methodology encompassing five sequential steps: embedding documents, reducing embeddings in dimensionality, clustering embeddings, tokenizing documents per cluster, and extracting the best representing words per topic.

However, with the emergence of Language Model-based approaches like Llama 2, there's potential for more refined topic representation. Directly analyzing all documents using Llama 2 may not be computationally feasible. While vector databases can aid in search, determining which topics to search for remains a challenge.

Instead, we can utilize the clusters and topics generated by BERTopic and employ Llama 2 to refine and distil this information into a more accurate representation. By integrating the strengths of both approaches, we can enhance the interpretability and depth of topic analysis.

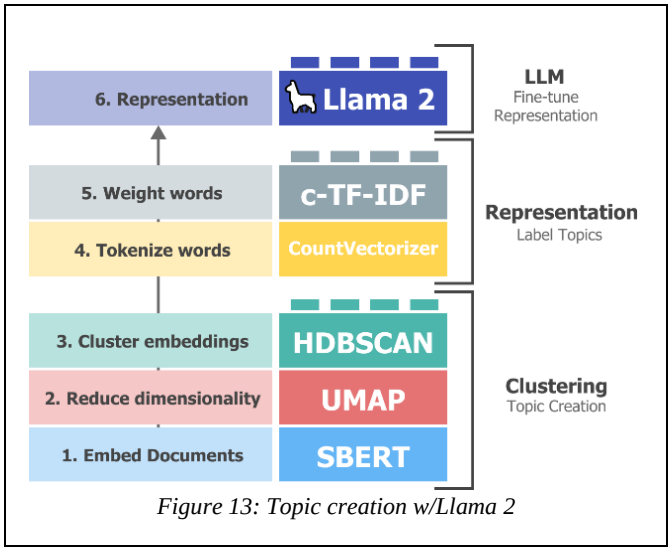

Figure 13: Topic creation w/Llama 2

References

Batuwita, R. and Palade, V. (2012). *CLASS IMBALANCE LEARNING METHODS FOR SUPPORT VECTOR MACHINES*. [online] Available at: https://www.cs.ox.ac.uk/people/vasile.palade/papers/Class-Imbalance-SVM.pdf. [4 January 2024]

Google Developers. (n.d.). *Imbalanced Data*. [online] Available at: https://developers.google.com/machine-learning/data-prep/construct/sampling-splitting/imbalanced-data. [4 February 2024]

Kim, J.H. (2019). Multicollinearity and misleading statistical results. *Korean Journal of Anesthesiology*, 72(6), pp.558–569. doi: https://doi.org/10.4097/kja.19087. [31 January 2024]