A Topic Modeling Approach: A Case Study Analysis of ESG Factors and SDGs in African Press

Mohamed El Ghali BENABOU

Master in Statistics and Econometrics

July 20, 2023 - January 20, 2024

Academic Supervisors:
Hafid Mouline
moulinehafid@gmail.com
Abdelhak Zoglat
a.zoglat@um5r.ac.ma

Company Mentors:
Imad Benelallam
imade@toum.ai
Naira Abdou Mohamed
naira@toum.ai

Dedication

This report is dedicated to my parents, whose unwavering love and support have been the foundation of my academic pursuits. Your sacrifices and encouragement have made this achievement possible. This achievement is as much yours as it is mine.

To my sisters, your love, encouragement, and unwavering belief in me have been a constant source of strength and inspiration. Thank you for always being there to lift me up and for cheering me on, even during the most challenging times.

To my mentors and teachers, who have inspired and guided me throughout my academic journey, thank you for your dedication to education and your commitment to my success.

And to my friends, who have stood by me through thick and thin, your friendship has been a source of comfort and motivation.

Acknowledgement

I would like to express my heartfelt gratitude to all those who have supported and guided me throughout the journey of completing this end of study report. First and foremost, I extend my deepest thanks to my supervisor, Abdou Mohamed Naira, whose expertise, insightful feedback, and unwavering encouragement have been invaluable. Your guidance has been instrumental in shaping this report.

I am also deeply thankful to my professors and lecturers at Mohammed V University for their knowledge and support throughout my academic journey. A special thanks to Professors Mouline Hafid and Zoglat Abdelhak for their particular insights and assistance.

I am also deeply grateful to my dear friend Omar Ettaouaje for generously sharing his ideas and insights. His perspective has been incredibly enriching to my work.

Special thanks to Abdessalam Bahafid for his invaluable assistance in overcoming coding challenges during my internship. His support and expertise were indispensable in navigating through complex technical issues.

Furthermore, I would like to acknowledge the immense contribution of Professor Imade Benelallam and my dear friend Nezar Bellazrak for providing me with the opportunity to learn and grow during my internship. Their guidance and mentorship have been pivotal in shaping my understanding and skills in the field.

My sincere appreciation goes to my family and friends for their constant support and encouragement. Your patience and understanding have been a source of strength for me.

Lastly, I am also thankful to my colleagues who were there alongside me during the internship, for their collaboration, camaraderie, and shared experiences. Your presence and support made the journey both productive and enjoyable. Thank you all for your invaluable contributions.

Abstract

This report investigates the discourse surrounding Environmental, Social, and Governance (ESG) factors and Sustainable Development Goals (SDGs) within African press data. Leveraging advanced topic modeling techniques with BERTopic, we aim to discern prevalent themes and patterns within the media narrative. Additionally, sentiment analysis is conducted on topics closely aligned with ESG and SDGs to understand their evolution over time and their perception within the media landscape.

The study begins by preprocessing a vast corpus of press articles sourced from diverse African publications. Through BERTopic, an innovative topic modeling algorithm based on transformer networks, we identify clusters of related topics, revealing the key issues and concerns driving ESG and SDGs discourse within African media.

Subsequently, we delve into sentiment analysis to gauge the prevailing attitudes towards topics closely associated with ESG and SDGs. By tracking sentiment shifts over time, we aim to uncover how perceptions of sustainability-related issues evolve within the media sphere.

The findings shed light on the multifaceted nature of ESG and SDGs discussions in African press, highlighting both the breadth of coverage and the nuances in sentiment. This study contributes to a deeper understanding of how sustainability issues are portrayed and perceived within African media, offering insights that can inform policy-making, corporate strategies, and societal engagement towards achieving sustainable development goals on the continent.

Keywords: ESG · SDGs · Africa · Topic Modeling · Sentiment Analysis · BERTopic

Résumé

Ce rapport examine le discours entourant les facteurs Environnementaux, Sociaux et de Gouvernance (ESG) et les Objectifs de Développement Durable (ODD) au sein des données de presse africaines. En utilisant des techniques avancées de modélisation de sujets avec BERTopic, nous visons à discerner les thèmes et motifs prédominants dans le récit médiatique. De plus, une analyse de sentiment est menée sur des sujets étroitement liés aux ESG et aux ODD afin de comprendre leur évolution au fil du temps et leur perception dans le paysage médiatique.

L'étude commence par le prétraitement d'un vaste corpus d'articles de presse provenant de diverses publications africaines. Grâce à BERTopic, un algorithme innovant de modélisation de sujets basé sur des réseaux de transformateurs, nous identifions des clusters de sujets liés, révélant les principales problématiques et préoccupations alimentant le discours sur les ESG et les ODD au sein des médias africains.

Ensuite, nous nous penchons sur l'analyse de sentiment pour évaluer les attitudes prédominantes à l'égard des sujets étroitement associés aux ESG et aux ODD. En suivant les évolutions de sentiment au fil du temps, nous visons à découvrir comment les perceptions des problématiques liées à la durabilité évoluent dans le domaine médiatique.

Les conclusions éclairent la nature multifacette des discussions sur les ESG et les ODD dans la presse africaine, mettant en évidence à la fois l'étendue de la couverture et les nuances de sentiment. Cette étude contribue à une meilleure compréhension de la manière dont les problématiques de durabilité sont présentées et perçues au sein des médias africains, offrant des perspectives qui peuvent éclairer l'élaboration de politiques, les stratégies d'entreprise et l'engagement sociétal en vue d'atteindre les objectifs de développement durable sur le continent.

Mots-clés: ESG · SDGs · Afrique · Topic Modeling · Analyse de sentiment · BERTopic

Contents

List of Tables

List of Figures

Introduction

Background and Motivation

The increasing global focus on sustainability has amplified the importance of Environmental, Social, and Governance (ESG) factors and Sustainable Development Goals (SDGs). These frameworks aim to address critical challenges such as climate change, social inequality, and corporate governance. In Africa, the media plays a pivotal role in shaping public discourse and influencing perceptions of these issues. Despite this, there is limited research on how ESG and SDGs are represented in African press.

This study seeks to fill this gap by leveraging advanced topic modeling techniques to analyze the discourse surrounding ESG and SDGs in African media. By understanding the prevalent themes and sentiments, this research aims to provide valuable insights for policymakers, corporations, and civil society organizations engaged in promoting sustainable development on the continent.

Scope and Research Objectives

The scope of this research encompasses an extensive analysis of press articles from various African publications, focusing on the portrayal of ESG factors and SDGs. The primary objectives of this study are:

- **To identify and categorize the main themes related to ESG and SDGs in African media** using the BERTopic modeling approach.
- **To analyze the sentiment associated with these themes**, examining the general attitudes towards sustainability-related topics.
- **To provide insights that can inform policy-making and corporate strategies**, enhancing the engagement and effectiveness of initiatives aimed at achieving sustainable development goals in Africa.

Company Background: ToumAI

ToumAI, a dynamic company born from AIOX Labs in June 2020, provided the fertile ground for the execution of my master thesis. With a core belief in technology's potential to foster inclusivity and sustainability, ToumAI emerged as a beacon of innovation within Africa's tech landscape.

ToumAI's mission to bridge data intelligence with societal impact resonates deeply with the objectives of my master thesis. Their commitment to inclusivity and sustainability aligns seamlessly with the thesis's exploration of ESG factors and SDGs within African

press coverage.

ToumAI provided the ideal backdrop for the execution of my master thesis. As a company at the forefront of innovation, ToumAI's ethos of leveraging technology for positive change mirrors the essence of the thesis's investigation into ESG factors and SDGs in African press.

Outline of the Research

This thesis is structured as follows:

The **Introduction** chapter provides an overview of the research topic, including background and motivation, scope, research objectives, and company background.

The **Literature Review** chapter delves into crucial aspects including Environment, Social, and Governance (ESG), exploring the intersection between Sustainable Development Goals (SDGs) and ESG. It also examines pertinent Machine Learning and Deep Learning techniques, along with Natural Language Processing methods, essential for developing a well-designed topic modeling approach.

The **Experimental Analysis and Discussion** chapter details the processes of data collection and preprocessing, provides an overview of the data, and presents the results of the topic modeling and sentiment analysis. It also discusses the limitations of the study.

The **Conclusion** chapter summarizes the main findings, and suggests directions for future research.

Chapter 1

Literature review

1.1 Environment, Social and Governance

ESG criteria, which stand for Environmental, Social, and Governance dimensions, are used to evaluate the impacts of a company on society and the environment. These criteria, although grounded in moral principles, are essential for measuring the sustainability and ethics of an investment or a company, thus forming the basis of responsible investment. The acronym "ESG" is widely used in the financial community to refer to these criteria, which typically constitute the pillars of non-financial analysis.

These criteria help assess a company's societal contribution in each dimension, thereby broadening the analysis of future financial performance, including profitability and risks.

In the context of ESG:

- Environment focuses on how a company manages its interactions with nature, such as reducing greenhouse gas emissions or preserving biodiversity.
- The social aspect concerns the company's practices towards its internal and external stakeholders, such as employees, customers, and local communities, with a focus on aspects like working conditions or community engagement.
- Governance evaluates how a company is directed and regulated, including board composition, financial transparency, and anti-corruption efforts.

Integrating ESG into a company's strategy can have numerous benefits, such as improved risk management, enhanced reputation, and long-term value creation for shareholders. Increasingly, investors are taking these criteria into account in their investment decisions, recognizing their significant economic impact.

Indeed, ESG is economic in that it directly influences the financial performance of companies and investors, going beyond philanthropic considerations. This integration helps reduce financial risks, improve operational performance, facilitate access to capital, and create sustainable value for all stakeholders.

1.2 SDGs and ESG

In September 2015, the 193 UN member states adopted the 2030 Agenda for Sustainable Development. It is a plan of action for people, for the planet, for prosperity, for peace and through partnerships. It sets out a vision for transforming our world by eradicating poverty and ensuring its transition to sustainable development. [1]

At the heart of this plan are the 17 Sustainable Development Goals (SDGs) (see Figure 1.1) , providing a clear policy framework for targeted actions at the national, regional, and international levels.

Fig. 1.1: UN's 17 Sustainable Development Goals [1]

The relationship between ESG (Environmental, Social, and Governance) and the SDGs is close and complementary. ESG criteria serve as a means of measurement and alignment to achieve these global sustainable development goals.

- Alignment with the SDGs: ESG criteria, covering environmental, social, and governance dimensions, are often aligned with the SDGs. For example, environmental criteria may be related to achieving SDGs concerning clean water, clean energy, or life on land. Similarly, social criteria can be associated with SDGs such as health, quality education, or gender equality.
- Contribution to the SDGs: Companies that integrate ESG practices directly contribute to achieving the SDGs. For example, by promoting diversity and inclusion in their workplace, they support SDGs on gender equality and decent work and economic growth.
- Reporting on the SDGs: Many companies now integrate the SDGs into their sustainability and corporate social responsibility (CSR) reports, alongside ESG criteria. This allows for better communication of their contribution to the SDGs and accountability for their progress in achieving them.
- Responsible investment: Investors who integrate ESG criteria often consider a company's alignment with the SDGs as an important factor. Companies positively contributing to the achievement of the SDGs may be favored in investment decisions.

ESG and the SDGs are interconnected, with the former providing a framework to assess the sustainability and social and environmental impact of companies, while the latter

represent a set of global goals for sustainable development. By integrating ESG criteria into their operations and investment decisions, companies can significantly contribute to the achievement of the SDGs and the construction of a more sustainable and equitable world for all.

Fig. 1.2: SDGs mapped to ESG targets [8]

1.3 Machine Learning and Deep Learning

Machine learning is a branch of artificial intelligence focused on developing algorithms that allow computers to learn from data and improve their performance on specific tasks over time. It involves the construction and training of mathematical models that can make predictions or decisions without being explicitly programmed for each task. Machine learning algorithms learn patterns and relationships within data, enabling them to make informed predictions or decisions in new situations. This technology has applications across a wide range of domains, including but not limited to, image and speech recognition, natural language processing, medical diagnosis, recommendation systems, and financial forecasting. As data availability continues to increase and computational power advances, machine learning continues to play an increasingly vital role in solving complex problems and driving innovation in various industries.

Deep learning is a subset of machine learning which focuses on the utilization of artificial neural networks, particularly deep neural networks with multiple layers, to automatically learn hierarchical representations of data. Its deep architectures enable the extraction of intricate patterns and features directly from raw data, eliminating the need for manual feature engineering. Leveraging techniques like backpropagation, deep learning models are trained iteratively, gradually improving their performance on tasks such as image recognition, natural language processing, and speech recognition. With scalability to handle large datasets and computational resources, deep learning has found widespread application across various domains, driving advancements in fields like computer vision, healthcare, and autonomous systems.

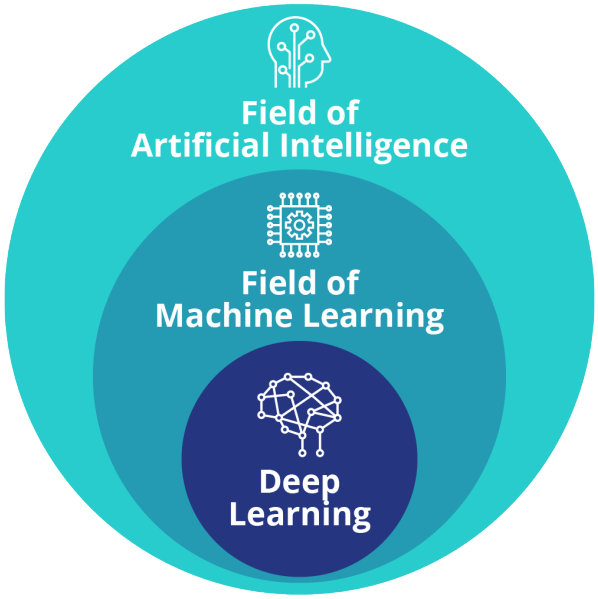


Fig. 1.3: Artificial Intelligence vs Machine Learning vs Deep Learning [25]

1.3.1 Neural Networks

A neural network (see Figure 1.4) is a computational model inspired by the structure and functioning of the human brain. It's composed of interconnected nodes, called neurons, organized in layers. Each neuron receives input signals, processes them, and produces an output signal which is passed on to other neurons. A neural network can be represented as a series of mathematical operations performed on input data. The most common neural network is called a feedforward neural network and is composed of three parts:

- Input Layer: The first layer of the neural network where the input data is fed.
- Hidden Layers: Intermediate layers between the input and output layers. Each neuron in these layers takes input from the previous layer, performs a transformation (typically a weighted sum of inputs followed by an activation function 1.5), and passes the output to the next layer.
- Output Layer: The final layer of the neural network. It produces the output based on the transformations that have occurred in the hidden layers.

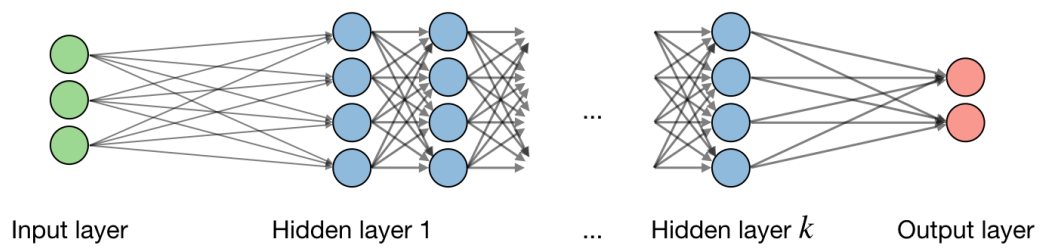


Fig. 1.4: Neural Network [5]

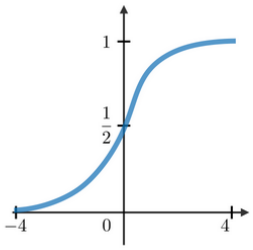
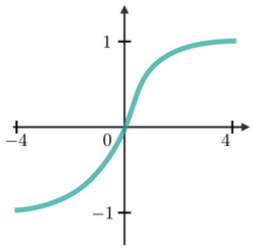
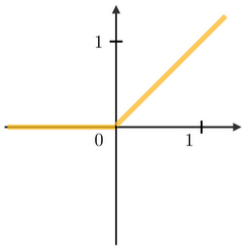
Sigmoid	Tanh	RELU
$g(z) = \frac{1}{1+e^{-z}}$	$g(z) = \frac{e^z - e^{-z}}{e^z + e^{-z}}$	$g(z) = \max(0, z)$
		

Fig. 1.5: Commonly used activation functions [5]

The neural network learns from iterating over multiple epochs (passes through the entire dataset) until the network learns to make accurate predictions through two propagation steps namely a forward propagation and a backpropagation.

1.3.1.1 Forward Propagation

Forward propagation is the process of computing outputs from inputs, layer by layer, until the final output is obtained. It's essentially the process of passing the input data through the network to get predictions.

Each neuron in a layer takes inputs from all neurons in the previous layer. It multiplies each input by a corresponding weight and sums them up. For neuron j in layer l, the weighted sum is computed as:

$$z_j^{(l)} = \sum_{i=1}^{n} w_{ji}^{(l)} \cdot a_i^{(l-1)} + b_j^{(l)}$$

Where:

- $w_{ji}(l)$ is the weight associated with the connection between neuron i in layer l-1 and neuron j in layer l.
- $a_{i(l-1)}$ is the output of neuron i in layer l-1.
- $b_{j(l)}$ is the bias term for neuron j in layer l.

The weighted sum is then passed through an activation function, which introduces non-linearity into the network. Common activation functions include sigmoid, tanh, ReLU (Rectified Linear Unit), etc. The output of the activation function becomes the output of the neuron:

$$a_j^{(l)} = \text{activation}(z_j^{(l)})$$

The output from each layer is used to calculate the output of the next layer until we get the output from the output layer. At that point, the forward propagation step for that iteration is complete.

1.3.1.2 Loss Function

A loss function measures how well a machine learning model's predictions match the actual target values. In the context of neural networks, the loss function quantifies the difference between the predicted output and the actual output for a given input. The goal is to minimize this difference during the training process.

One commonly used loss function in neural networks is the Mean Squared Error (MSE) loss function, particularly for regression problems. It calculates the average of the squared differences between the predicted values and the actual values:

$$\text{MSE} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$$

where:

- n is the number of samples in the dataset,
- y_i is the actual target value for the ith sample,
- $\hat{y}_i$ is the predicted value for the ith sample.

In binary classification problems, where the task is to classify inputs into one of two classes, a common loss function is the Binary Cross-Entropy Loss (also known as Log Loss). It measures the difference between the true binary label (0 or 1) and the predicted probability of the positive class:

$$\text{Binary Cross-Entropy Loss} = -\frac{1}{N}\sum_{i=1}^{N}\left(y_i \log(\hat{y}_i) + (1 - y_i)\log(1 - \hat{y}_i)\right)$$

where:

- N is the number of samples in the dataset,
- y_i is the true binary label (0 or 1) for the ith sample,
- $\hat{y}_i$ is the predicted probability of the positive class for the ith sample.

In multi-class classification problems, where the task is to classify inputs into one of more than two classes, the Cross-Entropy Loss (also known as Categorical Cross-Entropy

Loss) is commonly used. It measures the difference between the true class labels (represented as one-hot encoded vectors) and the predicted class probabilities:

$$\text{Cross-Entropy Loss} = -\frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{C} y_{ij}\log(\hat{y}_{ij})$$

where:

- N is the number of samples in the dataset,
- C is the number of classes,
- y_{ij} is the true probability that sample i belongs to class j (1 if sample i belongs to class j, 0 otherwise),
- $\hat{y}_{ij}$ is the predicted probability that sample i belongs to class j.

If the output involves multiple binary classification tasks, each task can be treated independently with its own binary cross-entropy loss. This setup is commonly known as multi-label classification.

For each binary classification task, you would use the binary cross-entropy loss. Then, the total loss for all tasks can be the sum of the individual binary cross-entropy losses:

$$\text{Total Loss} = \sum_{k=1}^{K} \text{Binary Cross-Entropy Loss}_k$$

The goal during training is to adjust the neural network's weights and biases to minimize the loss function. This is typically done using optimization algorithms like gradient descent, which iteratively update the parameters to move towards the minimum of the loss function.

1.3.1.3 Backpropagation

Backpropagation involves updating the weights and biases of the network to minimize the loss. Once the neural network generates its output, it's compared to the expected output using a loss function, quantifying the disparity between the predicted and actual results. Subsequently, the gradients of the loss function are utilized to modify the weights and biases through optimization techniques such as gradient descent.

The equations for updating the weights and biases during backpropagation using gradient descent are as follows:

$$W_{ij}^{(l)} := W_{ij}^{(l)} - \alpha\frac{\partial L}{\partial W_{ij}^{(l)}}$$
$$b_{i}^{(l)} := b_{i}^{(l)} - \alpha\frac{\partial L}{\partial b_{i}^{(l)}}$$

Where:

- $W_{ij}^{(l)}$ represents the weight of the connection between neuron i in layer $l-1$ and neuron j in layer l.
- $b_i^{(l)}$ represents the bias of neuron i in layer l.
- α is the learning rate.
- $\frac{\partial L}{\partial W_{ij}^{(l)}}$ and $\frac{\partial L}{\partial b_i^{(l)}}$ denote the gradients of the loss function L with respect to the weights and biases, respectively, in layer l.

1.3.2 Recurrent Neural Networks

Facing sequence learning problems like time series prediction, natural language processing, or speech recognition, standard neural networks may struggle due to varying input and output lengths and an inability to generalize features learned across different positions of the sequence.

Recurrent Neural Networks (RNNs) represent a class of artificial neural networks particularly adept at modeling sequential data. RNNs address these limitations by incorporating recurrent connections that enable them to retain information over time and share features learned across different positions of the sequence. Unlike traditional feedforward neural networks, which process data in a strictly one-directional manner, RNNs possess internal memory, allowing them to retain information about past inputs. This unique architecture enables RNNs to effectively capture temporal dependencies and patterns within sequential data, making them ideal for tasks such as time series prediction, natural language processing, speech recognition, and more.

At the core of an RNN lies the recurrent connection, which loops the network's hidden state back into itself, allowing information to persist over time. This recurrent feedback loop grants RNNs the ability to consider not only the current input but also past inputs encountered throughout the sequence. As a result, RNNs excel in tasks where context and temporal dynamics play crucial roles.

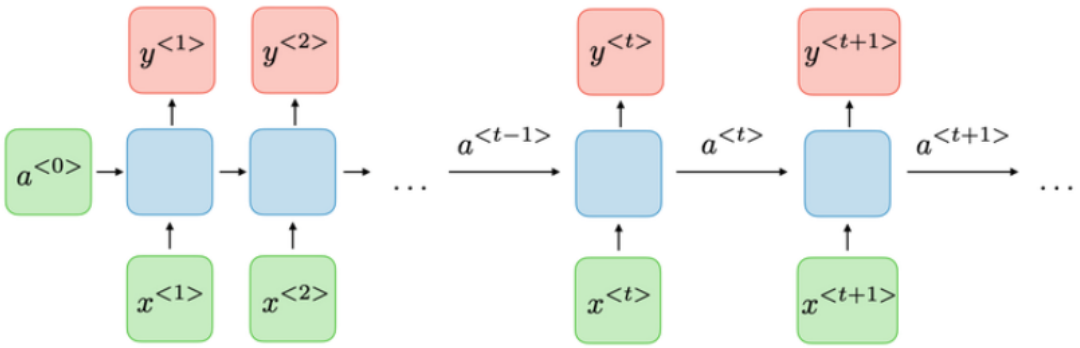


Fig. 1.6: Architecture of a traditional RNN [5]

1.3.2.1 RNN Types and use cases

Recurrent Neural Networks (RNNs) come in various configurations to handle different types of input-output relationships:

One-to-One : In this configuration, the RNN processes one input and generates one output at each time step. This setup is akin to traditional feedforward neural networks where each input corresponds to a single output without any temporal dependency. It's particularly useful in tasks where there's no need for sequential processing of inputs.

One-to-Many : Here, the RNN receives a single input and generates a sequence of outputs. This configuration is valuable for generating sequences of varying lengths based on a single input. For instance, in image captioning, a single image input can be transformed into a sequence of words describing it.

Many-to-One : This setup involves feeding a sequence of inputs to the RNN and obtaining a single output. Many-to-One configuration is commonly employed in tasks such as sentiment analysis of text. In sentiment analysis, the model processes a sequence of words and predicts the sentiment of the entire sentence.

Many-to-Many (Tx = Ty) : In this configuration, both input and output sequences have the same length. Many-to-Many setups are well-suited for sequence labeling tasks like part-of-speech tagging. Each input token is labeled with a corresponding tag, making it valuable in tasks requiring sequence labeling.

Encoder-Decoder (Sequence-to-Sequence) : The Encoder-Decoder architecture, also known as Sequence-to-Sequence, consists of two RNNs: an encoder and a decoder. The encoder processes the input sequence and produces a fixed-length context vector. Subsequently, the decoder takes this context vector and generates the output sequence. This architecture is commonly used in machine translation tasks, where the encoder processes the source language, and the decoder generates the target language.

Similar to regular neural networks, the training process of recurrent neural networks (RNNs) involves forward and backward propagation steps, referred to as the forward pass and backpropagation through time (BPTT).

1.3.2.2 Forward Pass

In the forward propagation step, for each timestep t, the activation $a^{<t>}$ and the output $y^{<t>}$ are expressed as follows:

$$a^{<t>} = g_1(W_{aa}a^{<t-1>} + W_{ax}x^{<t>} + b_a) \quad \text{and} \quad y^{<t>} = g_2(W_{ya}a^{<t>} + b_y)$$

where W_{ax}, W_{aa}, W_{ya}, b_a, b_y are coefficients that are shared temporally and g_1, g_2 activation functions.

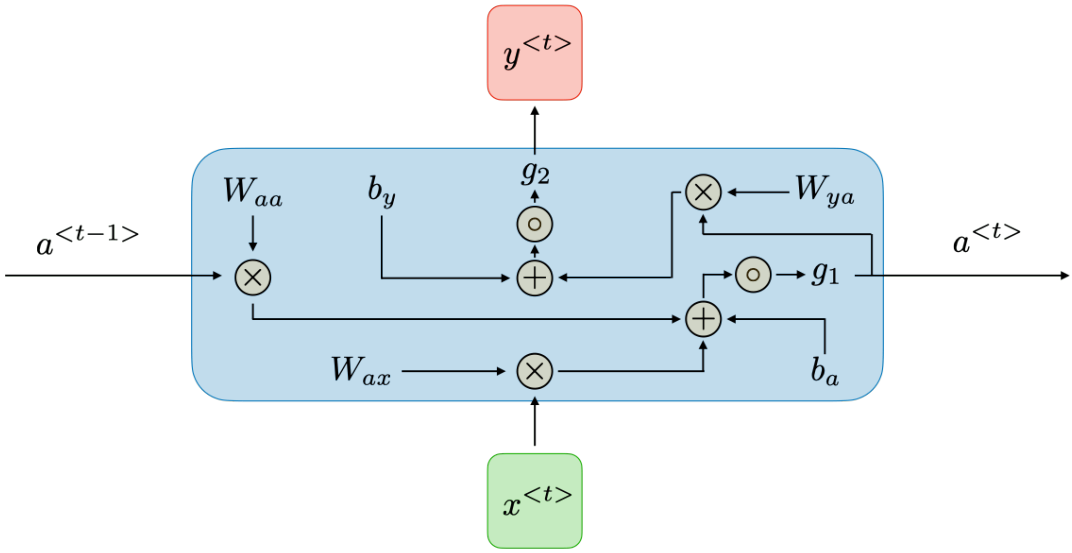


Fig. 1.7: Forward Propagation [5]

1.3.2.3 BackPropagation Through Time (BPTT)

In the case of a recurrent neural network, the loss function $\mathcal{L}$ of all time steps is defined based on the loss at every time step as follows:

$$\mathcal{L}(\hat{y}, y) = \sum_{t=1}^{T_y} \mathcal{L}(\hat{y}^{<t>}, y^{<t>})$$

The backpropagation is done at each point in time. At timestep T, the derivative of the loss $\mathcal{L}$ with respect to weight matrix W is expressed as follows:

$$\frac{\partial \mathcal{L}^{(T)}}{\partial W} = \sum_{t=1}^{T} \frac{\partial \mathcal{L}^{(T)}}{\partial W}\Big|_{(t)}$$

1.3.2.4 Exploding and vanishing Gradient

The exploding and vanishing gradient problems are common challenges encountered during the training of deep neural networks, particularly in recurrent neural networks (RNNs). These issues arise due to the nature of backpropagation, where gradients are calculated and propagated backward through the network to update the model's parameters.

The exploding gradient problem occurs when the gradients calculated during backpropagation become excessively large. This can lead to instability in training, causing the model's parameters to update in very large steps, which can result in oscillations or

divergence during training. On the other hand, the vanishing gradient problem occurs when the gradients become extremely small as they propagate backward through many layers of the network. This can cause the updates to the model's parameters to become negligible, hindering learning and leading to slow convergence or stagnation.

Gradient clipping (see Figure 1.8) is a technique used to address the exploding gradient problem by limiting the magnitude of gradients during training. By setting a threshold, gradients that exceed this threshold are scaled down to ensure that they do not become too large and destabilize the training process. It can also indirectly help alleviate the vanishing gradient problem to some extent by preventing gradients from becoming too small and allowing for more effective learning.

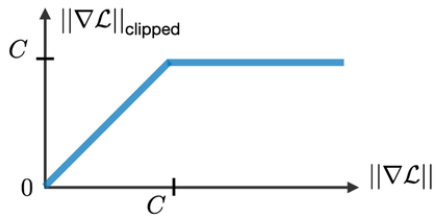


Fig. 1.8: Gradient clipping [5]

1.3.2.5 LSTM

While gradient clipping can prevent the gradients from becoming too small and potentially alleviate the vanishing gradient problem to some extent, it doesn't completely solve the underlying issue. Other techniques, such as using activation functions that are less prone to saturation (e.g., ReLU instead of sigmoid or tanh) or employing architectures specifically designed to address vanishing gradients Long Short-Term Memory (LSTM) [15] is a recurrent neural network architecture designed to address the challenges of the exploding and vanishing gradient problems. Unlike traditional RNNs, LSTM networks incorporate memory cells and gating mechanisms that enable them to retain information over long sequences and regulate the flow of gradients during training. This helps mitigate the vanishing gradient problem by allowing the network to learn long-term dependencies more effectively.

The LSTM architecture comprises four types of gates, typically represented by the symbol Γ, which play a pivotal role in recurrent neural networks. These gates are all defined by the equation:

$$\Gamma = \sigma(Wx^{<t>} + Ua^{<t-1>} + b)$$

Here, W, U, and b are coefficients specific to the gate, while σ denotes the sigmoid function. The Gamma gates serve various functions within the network:

- The Update gate (Γ_u) determines how much importance should be given to past information in the current time step.

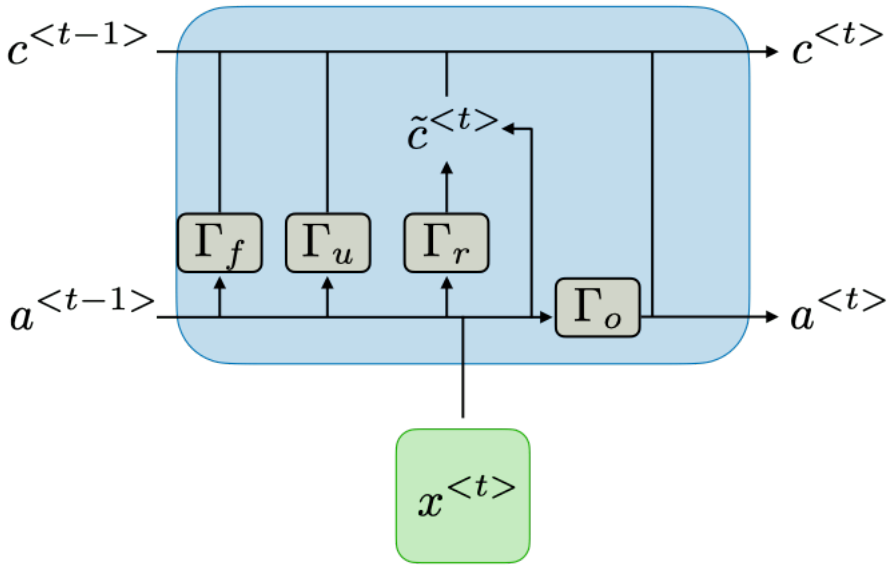


Fig. 1.9: The Long Short-Term Memory (LSTM) architecture [5]

- The Relevance gate (Γ_r) controls whether previous information should be retained or dropped.
- The Forget gate (Γ_f) decides whether to erase information stored in a memory cell.
- The Output gate (Γ_o) regulates how much information from the memory cell should be revealed to the next layer or output.

These gates collectively contribute to the network's ability to retain and update information over time, making them essential components of recurrent neural network architectures. The evolution of these gates is encapsulated in three equations governing parameter updates:

$$\begin{aligned}
\tilde{c}^{<t>} &= \tanh(W_c[\Gamma_r \star a^{<t-1>}, x^{<t>}] + b_c) \\
c^{<t>} &= \Gamma_u \star \tilde{c}^{<t>} + \Gamma_f \star c^{<t-1>} \\
a^{<t>} &= \Gamma_o \star c^{<t>}
\end{aligned}$$

Here $\tilde{c}^{<t>}$ is the candidate memory cell value at time step t, $c^{<t>}$ is the actual memory cell value at time step t and $a^{<t>}$ is the output activation at time step t.

1.3.2.6 BRNN

RNNs are powerful models for sequential data processing. However, they suffer from a fundamental drawback: they process input sequences in a strictly forward direction, which limits their ability to effectively capture information from both past and future contexts simultaneously.

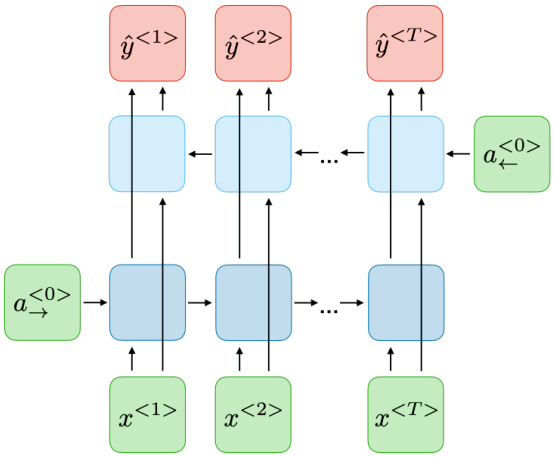


Fig. 1.10: Bidirectional RNN (BRNN)[5]

This unidirectional processing can lead to a bias where the model heavily relies on the preceding context and might overlook crucial information from the future context, especially in tasks where bidirectional information is essential for accurate predictions. For instance, in language understanding tasks, the meaning of a word often depends not only on the preceding words but also on the subsequent ones.

To address this limitation, researchers proposed Bidirectional Recurrent Neural Networks (BRNNs), which allow information to flow in both forward and backward directions through the sequence (see Figure 1.10). By processing the input sequence in both directions, BRNNs can capture context from both past and future states, enabling them to better understand and model dependencies in sequential data.

1.3.3 Transformers

In the landscape of artificial intelligence and natural language processing (NLP), Transformers stand as a revolutionary architecture that has reshaped the way machines comprehend and generate human language. Originally introduced in the seminal paper "Attention is All You Need"[26], Transformers have rapidly become the cornerstone of many state-of-the-art NLP models due to their remarkable ability to capture long-range dependencies and contextual information within sequences of data.

Unlike traditional sequence models such as recurrent neural networks (RNNs) and convolutional neural networks (CNNs), Transformers rely solely on self-attention mechanisms to weigh the importance of different words or tokens in a sequence. This approach allows Transformers to concurrently process entire sequences, rendering them notably more efficient and scalable than LSTM. Furthermore, Transformers excel in capturing contextual information by considering all input tokens simultaneously, mitigating the vanishing gradient problem associated with RNNs and enabling more effective modeling of long-range dependencies.

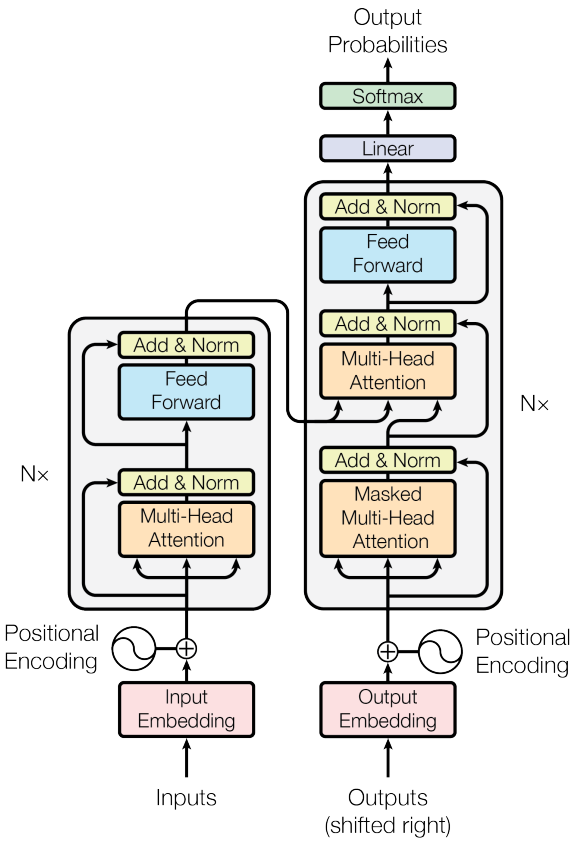


Fig. 1.11: The encoder-decoder structure of the Transformer architecture[26]

1.3.3.1 Word embedding

In Transformers, embeddings are used to represent tokens in a continuous vector space, allowing the model to operate on tokenized input sequences. Each token in the input sequence is associated with an embedding vector, which captures semantic and syntactic information about the token.

Mathematically, let $X = \{x_1, x_2, ..., x_n\}$ be the input sequence of tokens, and let E be the embedding matrix. The embedding vector e_i for token x_i is obtained by looking up the corresponding row in the embedding matrix:

$$e_i = E[x_i]$$

The embedding matrix E is typically initialized randomly or pre-trained using techniques such as Word2Vec, GloVe, or contextualized embeddings like those from BERT. These embeddings serve as the initial input to the Transformer model before further processing through the self-attention mechanism and subsequent layers.

1.3.3.2 Positional encoding

Positional encoding is then used to inject information about the position of tokens in a sequence into the model. Since Transformers do not inherently understand the sequential order of tokens, positional encoding is crucial for helping the model distinguish between different positions in the input sequence.

There are various ways to implement positional encoding, but one common method is to use sine and cosine functions. This method was introduced in the original Transformer paper "Attention is all you need"[26].

In positional encoding, a unique encoding vector is assigned to each position in the input sequence. These encoding vectors are added to the input embeddings of the tokens before feeding them into the model. This allows the model to learn positional information along with the token semantics.

The positional encoding for each position is calculated as follows:

$$\mathrm{PE}_{\mathrm{pos},2i} = \sin\left(\frac{\mathrm{pos}}{10000^{2i/d_{\mathrm{model}}}}\right)$$
$$\mathrm{PE}_{\mathrm{pos},2i+1} = \cos\left(\frac{\mathrm{pos}}{10000^{2i/d_{\mathrm{model}}}}\right)$$

Where:

- pos is the position of the token.
- i is the dimension of the positional encoding vector.
- d_{model} is the dimensionality of the model (embedding dimension).
- $\mathrm{PE}_{\mathrm{pos},2i}$ represents the i-th dimension of the positional encoding vector for position pos using sine function.
- $\mathrm{PE}_{\mathrm{pos},2i+1}$ represents the i-th dimension of the positional encoding vector for position pos using cosine function.

The choice of $10000^{2i/d_{\mathrm{model}}}$ inside the sine and cosine functions allows the model to learn different frequencies for each dimension of the positional encoding. This use of sine and cosine functions ensures that the positional encoding is continuous and can represent various positions accurately. Consequently, the positional encoding vectors are added to the input embeddings, allowing the model to learn both the semantics of tokens and their positional information.

1.3.3.3 Layer Normalization

Layer Normalization (LN) is a technique used to normalize the activations of each layer in a neural network. It's similar to Batch Normalization but operates on a per-feature

basis instead of per-batch [6]. In the context of transformers, it is typically applied after the multi-head self-attention and feedforward layers.

Mathematically, for a given layer, the layer normalization operation can be defined as follows:

Given input vector $x = (x_1, x_2, ..., x_n)$ where n is the number of features (or neurons) in that layer, the layer normalization operation can be described as:

$$\mathrm{LN}(x) = \gamma \frac{x - \mu}{\sigma} + \beta$$

where:

- μ is the mean of the input vector x,
- σ is the standard deviation of x,
- γ is a learnable scaling parameter,
- β is a learnable shift parameter.

The mean and standard deviation are computed over the feature dimensions. Essentially, layer normalization ensures that the mean of each feature is zero and the variance is one, which helps stabilize the training process.

1.3.3.4 Residual connections

Residual connections (see Figure 1.12), drawing inspiration from skip connections, serve as a remedy for the vanishing gradient issue and contribute to the effective training of deep networks. They are introduced by adding the input of a certain layer to its output before applying a non-linear activation function By allowing the model to learn both the original representation and the residual, they promote smoother optimization processes and enhance the flow of gradients.

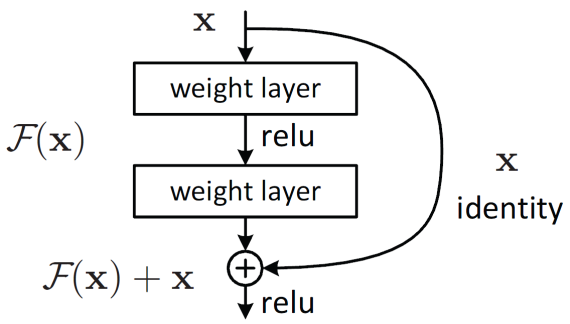


Fig. 1.12: Residual Connections [14]

1.3.3.5 Self-Attention

The self-attention mechanism is a pivotal component of the Transformer architecture, empowering the model to evaluate the significance of each token's context within an input sequence. Within each attention unit, the Transformer model learns three weight matrices: the query weights W_Q, the key weights W_K, and the value weights W_V. For every token i, the input token representation x_i undergoes multiplication with each of these weight matrices, yielding a query vector $q_i = x_i W_Q$, a key vector $k_i = x_i W_K$, and a value vector $v_i = x_i W_V$.

Mathematically, the self-attention mechanism can be denoted as follows:

Given an input sequence of tokens $X = \{x_1, x_2, ..., x_n\}$, where n is the sequence length, the attention score α_{ij} between token x_i and token x_j is calculated as:

$$\alpha_{ij} = \text{softmax}\left(\frac{q_i k_j^T}{\sqrt{d_k}}\right)$$

Here, $q_i = x_i W_Q$ and $k_j = x_j W_K$, and d_k represents the dimensionality of the key vectors. The softmax function guarantees that the attention scores sum up to 1 across all tokens in the sequence, forming a probability distribution over the tokens.

Subsequently, the attention scores are utilized to compute a weighted sum of the value vectors V, which encompass information regarding each token's context:

$$\text{Attention}(X) = \sum_{j=1}^{n} \alpha_{ij} v_j$$

This weighted sum represents the contextualized representation of token x_i based on its attention to other tokens in the sequence.

1.3.3.6 Multi-head attention

An ensemble of matrices (W_Q, W_K, W_V) constitutes an attention head, with multiple attention heads present in each layer of a Transformer model. While each attention head focuses on the relevant tokens for a given token, the utilization of multiple attention heads enables the model to discern various interpretations of "relevance." Furthermore, the concept of relevance can expand progressively across layers. Many attention heads within Transformers capture relevance patterns that resonate with human understanding. For instance, certain attention heads might prioritize neighboring words, while others emphasize relationships between verbs and their direct objects. The computations for each attention head can be executed simultaneously, facilitating efficient processing. The outputs of the attention layer are concatenated and passed to subsequent feed-forward neural network layers.[29]

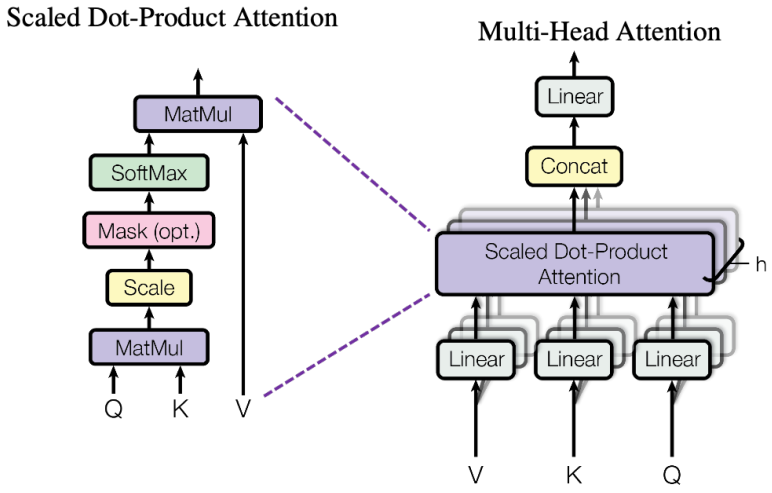


Fig. 1.13: Self-Attention [26]

Specifically, if we index the multiple attention heads with i, we express:

$$\text{MultiheadedAttention}(Q, K, V) = \text{Concat}_{i \in [\#heads]}(\text{Attention}(XW_i^Q, XW_i^K, XW_i^V))W^O$$

Here, X represents the concatenation of word embeddings, W_i^Q, W_i^K, and W_i^V denote projection matrices unique to each attention head indexed by i, and W^O signifies the final projection matrix owned by the collective multi-headed attention mechanism.

1.3.3.7 Masked attention

In certain scenarios, it becomes imperative to sever attention links between specific word-pairs. For instance, in a decoder where token t ought not to access token $t+1$, this task is achieved prior to the softmax stage by introducing a mask matrix M. Within M, entries corresponding to attention links requiring termination are assigned a value of $-\infty$, while all other entries remain 0:

$$\text{MaskedAttention}(Q, K, V) = \text{softmax}\left(M + \frac{QK^T}{\sqrt{d_k}}\right)V$$

As an illustration, consider the mask matrix utilized in autoregressive modeling:

$$M = \begin{bmatrix} 0 & -\infty & -\infty & \dots & -\infty \\ 0 & 0 & -\infty & \dots & -\infty \\ 0 & 0 & 0 & \dots & -\infty \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & 0 \end{bmatrix}$$

Essentially, this arrangement signifies that each token retains the ability to attend to itself and all preceding tokens, but is restricted from attending to subsequent tokens.

1.3.3.8 The encoder structure

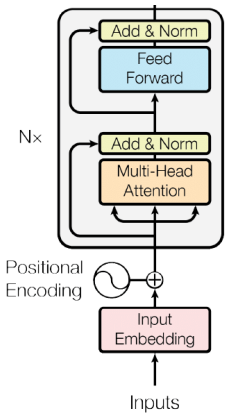


Fig. 1.14: The Transformer encoder structure [26]

The encoder (refer to Figure 1.14) takes an input sequence, such as a sentence, and transforms it into a sequence of hidden representations. This process begins with the input embeddings, which map each token in the sequence to a high-dimensional vector space. These embeddings capture semantic and syntactic information about the tokens.

Next, the positional encoding is added to the input embeddings. Positional encoding helps the model understand the order of tokens in the sequence, crucial for tasks like language understanding or translation. It introduces positional information into the embeddings using sine and cosine functions.

The encoded sequence then undergoes a series of transformer blocks. Each transformer block consists of a self-attention mechanism followed by position-wise feedforward neural networks. The self-attention mechanism allows each token in the sequence to attend to other tokens, capturing dependencies and relationships within the sequence. Additionally, multi-head attention enables the model to focus on different aspects of the input simultaneously, enhancing its ability to capture diverse patterns.

Normalization and residual connections are employed within each transformer block to stabilize the training process and facilitate gradient flow. Normalization techniques like layer normalization help in maintaining a consistent distribution of values throughout the network, while residual connections enable effective information flow across layers.

The output of the encoder is a sequence of context-rich representations, where each token is enriched with information from its surrounding tokens.

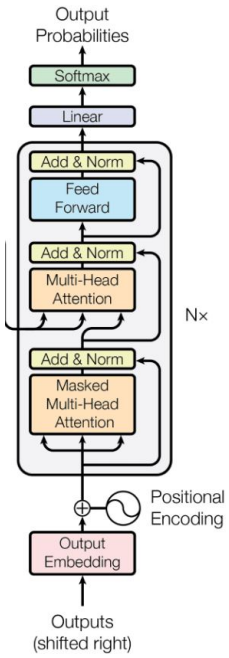


Fig. 1.15: The Transformer decoder structure [26]

1.3.3.9 The decoder structure

The decoder (refer to Figure 1.15), on the other hand, takes the encoded representations produced by the encoder and generates an output sequence token by token. Similar to the encoder, the decoder also starts with input embeddings and positional encodings.

However, in addition to self-attention, the decoder employs another attention mechanism called masked attention. Masked attention prevents the decoder from attending to future tokens during the generation process, ensuring causal generation and maintaining the autoregressive property of the model.

The decoder also incorporates encoder-decoder attention (see Figure 1.16), where each token in the output sequence attends to the encoded representations produced by the encoder. This enables the decoder to leverage information from the input sequence while generating the output sequence, facilitating tasks like sequence-to-sequence translation.

Similar to the encoder, the decoder consists of multiple transformer blocks, each containing self-attention, masked attention, and position-wise feedforward layers. Normalization and residual connections are applied within these blocks to ensure stable training and effective information flow.

1.3.3.10 The global architecture

The global transformer architecture comprises an encoder and a decoder, both interconnected through attention mechanisms. In this architecture, the encoder processes the input sequence, producing a set of encoded representations one for every encoder layer. These

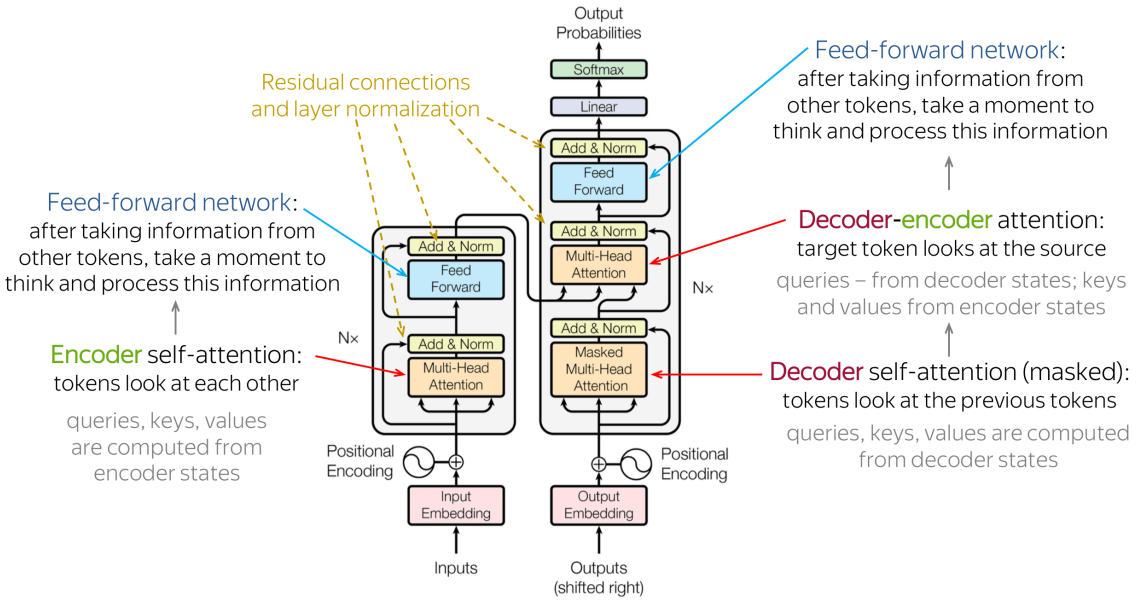


Fig. 1.16: The Transformer architecture detailed [4]

representations are then passed to the decoder, where they serve as crucial information for generating the output sequence.

During the decoding process, the decoder attends to the encoded representations produced by the encoder, utilizing both self-attention and encoder-decoder attention mechanisms. This bidirectional flow of information enables the model to capture dependencies between input and output sequences effectively. It facilitates tasks such as machine translation and text generation by allowing the model to understand the context of the input sequence and generate coherent output sequences accordingly.

1.3.4 BERT

The BERT model was proposed in the paper "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding" by Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT is a bidirectional transformer pretrained using a combination of masked language modeling and next sentence prediction on a large corpus comprising the Toronto Book Corpus and Wikipedia. After pre-training, BERT can be fine-tuned on specific downstream tasks with relatively small amounts of data, achieving state-of-the-art performance on a wide range of NLP benchmarks.

Traditional models like GPT (Generative Pre-trained Transformer) are unidirectional, meaning they only consider context from left to right. In contrast, BERT uses bidirectional training to read the entire sequence of words at once, allowing it to understand the context more comprehensively.

BERT's primary objective is to perform various NLP tasks such as sentiment analysis, question answering, and named entity recognition. By using the encoder part of the Transformer model, BERT captures both semantic and syntactic information within its

embeddings, which is crucial for these diverse tasks.

Unlike models designed for text generation or translation, BERT does not employ the decoder component of the Transformer. Instead, BERT excels at understanding and encoding text into rich, meaningful embeddings. This characteristic necessitates further processing of BERT's output for various applications. For example, techniques such as cosine similarity can be used to compare embeddings and derive similarity scores between texts.

1.3.4.1 Training

BERT is pretrained on large text corpora, including the Toronto Book Corpus and Wikipedia, using a large model (12-layer to 24-layer Transformer) over an extensive period (1 million update steps). Pre-training is resource-intensive but is performed only once per language.

Masked Language Modeling (MLM) One of BERT's core techniques is masked language modeling. During pre-training, 15% of the words in the input text are randomly masked, and the model is tasked with predicting these masked words. This trains the model to understand the context of a word based on both its preceding and following words. The MLM objective can be formally defined as minimizing the cross-entropy loss between the predicted words and the actual words.

Next Sentence Prediction (NSP) To further enhance its understanding of context, BERT is also trained on a task called next sentence prediction. This involves taking pairs of sentences and training the model to predict whether the second sentence follows the first in the original text. This binary classification task helps the model understand the relationship between sentences. The NSP objective is minimized using cross-entropy loss as well.

1.3.4.2 Fine-tuning

After pre-training, BERT can be fine-tuned on specific downstream tasks with relatively small amounts of data. Fine-tuning is efficient and allows BERT to achieve state-of-the-art results on various NLP tasks, including sentence classification, sentence-pair classification, named entity recognition, and question answering.

Google has released several pre-trained BERT models, enabling most NLP researchers to leverage these models directly without needing to pre-train from scratch.

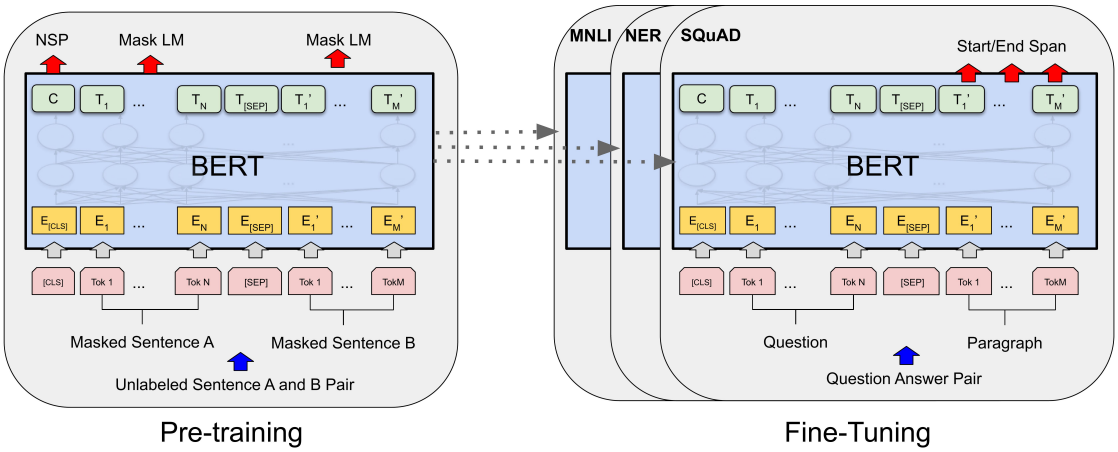


Fig. 1.17: Pretraining and Fine-Tuning of BERT [9]

1.3.5 MTEB

The Massive Text Embedding Benchmark (MTEB) provides a comprehensive evaluation framework for text embedding models across eight tasks (refer to Figure 1.18):

- Bitext Mining: Match sentences from two different languages, typically translations, using cosine similarity. Metrics include F1, accuracy, precision, and recall.
- Classification: Train a logistic regression classifier on embedded train sets and score it on test sets. Metrics include accuracy, average precision, and F1.
- Clustering: Group sentences or paragraphs into clusters using mini-batch k-means. Evaluate using v-measure, which is unaffected by label permutations.
- Pair Classification: Assign labels to pairs of text inputs (e.g., duplicate or paraphrase) based on various distance metrics. Main metric is average precision score based on cosine similarity.
- Reranking: Rank results according to relevance to a query using cosine similarity. Metrics include mean MRR@k and MAP, with MAP as the main metric.
- Retrieval: Find relevant documents from a corpus given queries. Metrics include nDCG@k, MRR@k, MAP@k, precision@k, and recall@k, with nDCG@10 as the main metric.
- Semantic Textual Similarity (STS): Determine similarity between sentence pairs using various distance metrics. Evaluate using Pearson and Spearman correlations, with Spearman correlation based on cosine similarity as the main metric.
- Summarization: Score machine-generated summaries against human-written summaries using cosine similarity. Evaluate using Pearson and Spearman correlations, with Spearman correlation based on cosine similarity as the main metric.

MTEB emphasizes diversity, simplicity, extensibility, and reproducibility, with a wide range of datasets covering multiple languages and text lengths. Its open-source nature

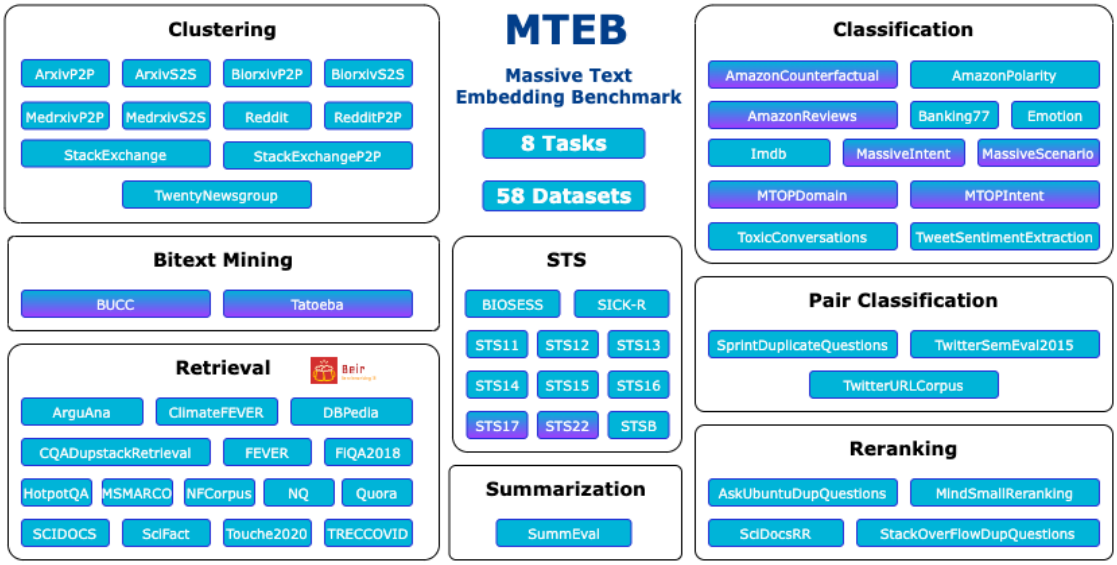


Fig. 1.18: An overview of tasks and datasets in MTEB. Multilingual datasets are marked with a purple shade [20]

enables easy benchmarking of various models, contributing to the selection of appropriate embedding models for diverse real-world applications.

1.3.6 PCA

Principal Component Analysis (PCA) is a powerful statistical method used for dimensionality reduction and feature extraction in data analysis. Its primary purpose is to simplify complex datasets by transforming them into a lower-dimensional space while preserving the essential information contained in the original data. PCA achieves this by identifying the directions, or principal components, along which the data varies the most, thus capturing the maximum variance in the dataset.

PCA is commonly employed to reduce the number of variables in high-dimensional datasets, visualize data in lower-dimensional spaces for easier interpretation, identify patterns and relationships in data, remove noise and redundancy from datasets and prepare data for subsequent analysis or modeling tasks.

Principal Component Analysis (PCA) relies on eigenvalue decomposition, a fundamental concept in linear algebra. Given a covariance matrix Σ calculated from the centered data, PCA aims to find its eigenvectors $\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_d$ and corresponding eigenvalues $\lambda_1, \lambda_2, \ldots, \lambda_d$, where d is the dimensionality of the data. The eigenvectors represent the principal directions (components) of the data, and the eigenvalues indicate the variance captured by each component. The covariance matrix Σ is diagonalized as $\Sigma = V\Lambda V^T$, where V is a matrix whose columns are the eigenvectors and Λ is a diagonal matrix containing the eigenvalues. The principal components are then obtained by projecting the centered data onto the eigenvectors: $\mathbf{Z} = XV$, where X is the centered data matrix and $\mathbf{Z}$ contains the lower-dimensional representations of the data. By selecting the top k eigenvectors corresponding to the largest eigenvalues, PCA effectively reduces the

dimensionality of the data while retaining the maximum variance.

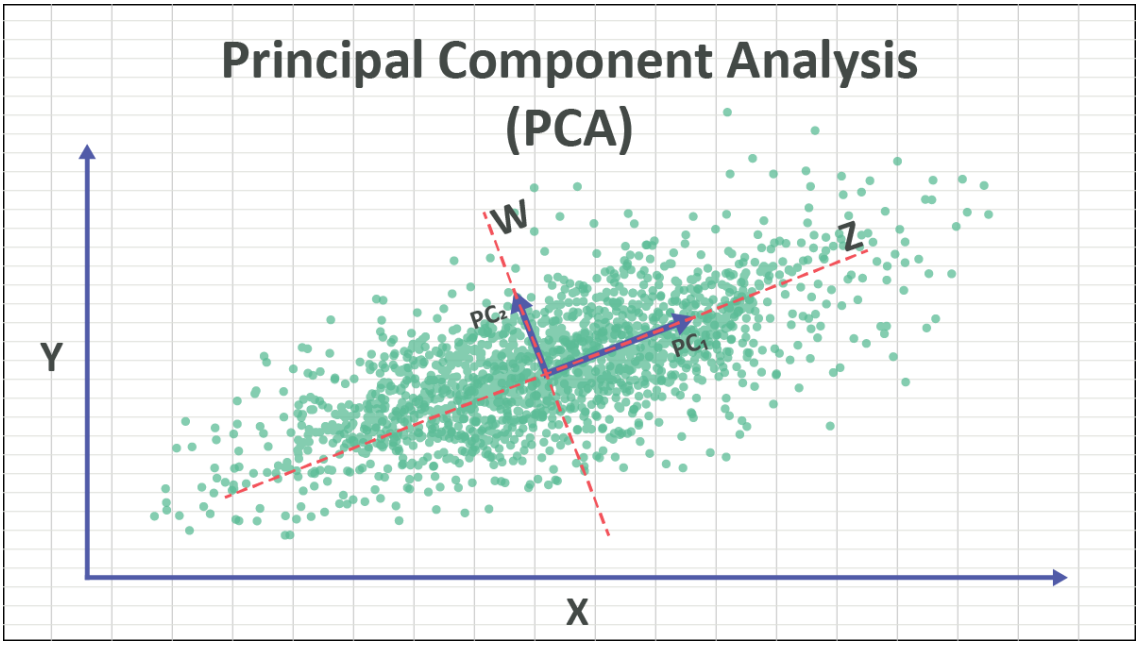


Fig. 1.19: Principal Component Analysis on a 2-Dimensional dataset [2]

PCA is a versatile technique with several strengths that make it invaluable in data analysis. One of its primary benefits is its ability to effectively reduce the dimensionality of high-dimensional datasets while retaining the most critical variance. This simplification makes it easier to visualize and interpret complex data structures, aiding in data exploration and understanding. Moreover, PCA operates as an unsupervised technique, eliminating the need for labeled data and thereby broadening its applicability across various datasets. Additionally, PCA assists in uncovering underlying patterns and relationships within the data, enhancing insights and decision-making processes.

However, despite its advantages, PCA does have limitations to consider. One notable assumption of PCA is the presence of linear relationships between variables, which may not always hold true in real-world datasets with intricate, nonlinear structures. This limitation can affect the accuracy and reliability of PCA results, particularly in datasets with complex interactions between variables. Furthermore, PCA's performance can be sensitive to the scale of the variables, potentially introducing bias and affecting the outcomes. Additionally, interpreting the principal components generated by PCA can be challenging, especially in high-dimensional spaces where the relationships between variables are less intuitive, requiring careful analysis and domain knowledge.

1.3.7 UMAP

Uniform Manifold Approximation and Projection (UMAP) is a recently proposed method for manifold learning that aims to accurately represent local structure while also incorporating global structure more effectively . Compared to t-SNE, UMAP offers several advantages. It has demonstrated better scalability with large datasets, whereas t-SNE often struggles with such datasets.

UMAP operates based on three key hypotheses:

1. The data is uniformly distributed on a Riemannian manifold.
2. The Riemannian metric is locally constant.
3. The manifold is locally connected.

These assumptions enable UMAP to represent the manifold using a fuzzy topological structure of high-dimensional data points. The process involves searching for a fuzzy topological structure of low-dimensional projections of the data.

To construct the fuzzy topological structure, UMAP represents the data points using a high-dimensional graph (see Figure 1.20 A), where edge weights indicate the likelihood of connection between two points. UMAP employs an exponential probability distribution to compute the similarity between high-dimensional data points:

$$p_{i|j} = \exp\left(-\frac{d(x_i, x_j) - \rho_i}{\sigma_i}\right)$$

where $d(x_i, x_j)$ is the distance between the i-th and j-th data points, and ρ_i is the distance between the i-th data point and its first nearest neighbor. In cases where the graph's weight between nodes i and j is not equal to the weight between nodes j and i, UMAP uses a symmetrization of the high-dimensional probability:

$$p_{ij} = p_{i|j} + p_{j|i} - p_{i|j}p_{j|i}$$

UMAP specifies the number of nearest neighbors (k) as:

$$k = 2^{\sum_i p_{ij}}$$

Once the high-dimensional graph is constructed, UMAP optimizes the layout of a low-dimensional analogue to closely resemble it (see Figure 1.20 B). For modeling distance in low dimensions, UMAP employs a probability measure similar to the Student t-distribution:

$$q_{ij} = \left(1 + a(y_i - y_j)^{2b}\right)^{-1}$$

where $a \approx 1.93$ and $b \approx 0.79$ for default UMAP settings. UMAP uses binary cross-entropy (CE) as a cost function due to its ability to capture the global data structure:

$$CE(P, Q) = \sum_i \sum_j [p_{ij} \log\left(\frac{p_{ij}}{q_{ij}}\right) + (1 - p_{ij}) \log\left(\frac{1 - p_{ij}}{1 - q_{ij}}\right)]$$

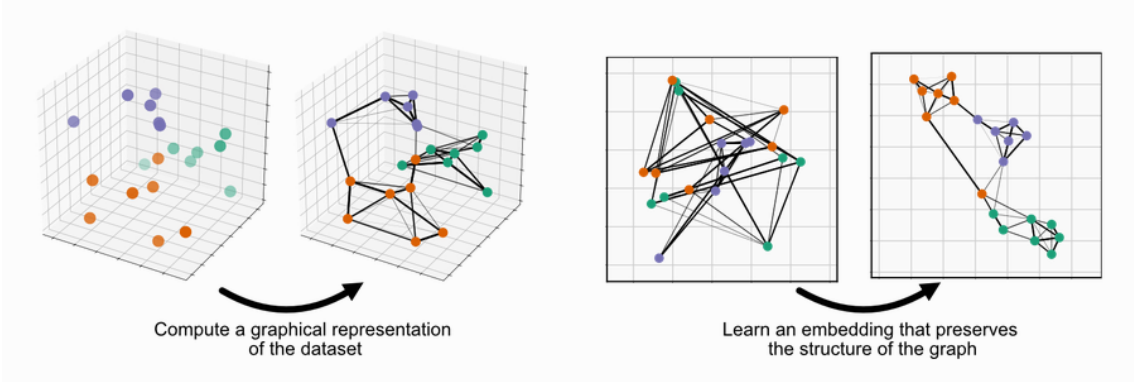


Fig. 1.20: UMAP steps [18]

Here, P represents the probabilistic similarity of the high-dimensional data points, and Q represents the low-dimensional data points.

UMAP employs the derivative of the cross-entropy to update the coordinates of the low-dimensional data points in order to optimize the projection space until convergence. Stochastic Gradient Descent (SGD) is utilized for its faster convergence and reduced memory consumption, as gradients are computed for a subset of the dataset.

UMAP's performance is influenced by several hyperparameters:

- The dimensionality of the target embedding.
- The number of neighbors (k), where a smaller value captures very local interpretations with fine detail structure, while a larger value estimates based on larger regions, potentially missing some fine detail structure.
- The minimum allowed distance between points in the embedding space, with lower values capturing the true manifold structure more accurately but potentially resulting in dense clouds that hinder visualization.

Typically, we validate UMAP results using metrics from downstream tasks. For instance, in classification scenarios, we might employ objective metrics like the F1-Score to assess dimensionality reduction performance. However, achieving a high F1-Score doesn't guarantee that UMAP has accurately represented the data's structure. A high accuracy score in the downstream task simply indicates that the data can be separated in lower dimensions and performs well based on its inputs.[11]

Trustworthiness [27] is a metric we can use to assess the preservation of the underlying data's structure. It measures the extent to which the local structure is maintained and falls within the range of [0, 1]. It is defined as:

$$T(k) = 1 - \frac{2}{nk(2n - 3k - 1)} \sum_{i=1}^{n} \sum_{j \in \mathcal{N}_i^k} \max(0, (r(i,j) - k))$$

where for each sample i, $\mathcal{N}_i^k$ are its k nearest neighbors in the output space, and every sample j is its $r(i,j)$-th nearest neighbor in the input space. In other words, any

unexpected nearest neighbors in the output space are penalised in proportion to their rank in the input space.

1.3.8 HDBSCAN

Density-Based Spatial Clustering of Applications with Noise (DBSCAN) is a clustering algorithm that groups together points in a dataset based on their density. It defines clusters as dense regions of points separated by regions of lower density, allowing it to identify clusters of arbitrary shape. The key parameters in DBSCAN are the minimum number of points (MinPts) required to form a cluster and a distance threshold (ε) that determines the neighborhood size for density calculations.

Hierarchical DBSCAN* (HDBSCAN) is a clustering algorithm constructed upon an adapted version of DBSCAN, known as DBSCAN*, where border points are classified as noise. In contrast to DBSCAN*, HDBSCAN doesn't determine clusters using a universal epsilon threshold. Instead, it establishes a hierarchy considering all potential epsilon values relative to a minimum cluster size defined by minPts.[17]

HDBSCAN operates through several key steps to identify clusters in data. Initially, the space is transformed based on density or sparsity considerations. Next, a minimum spanning tree is constructed for the distance-weighted graph. Following this, a cluster hierarchy is formed from the connected components of this tree. This hierarchy is then condensed by considering a minimum cluster size criterion. Finally, stable clusters are extracted from the condensed tree, representing the meaningful clusters in the data.

For a proper formulation of these steps with regards to a value of mpts (minimum points parameter), the author of HDBSCAN defined the notions of Core Object, Core Distance, σ-Core Object, and Mutual Reachability Distance:

Core Object Given parameters ε and k, an object is considered a core object if its ε-neighborhood contains at least k objects.

Core Distance The core distance of an object x with regards to a parameter mpts denoted $core_k(x)$ represents the distance at which x becomes a core object when considering its k-nearest neighbors.

ε-Core Object An object x is called an ε-core object with regards to k if $core_k(x) \leq \varepsilon$.

Mutual Reachability Distance The mutual reachability distance between two objects x and y with regards to k is define as:

$$\text{m-reach}_k(x, y) = \max(core_k(x), core_k(y), d(x, y))$$

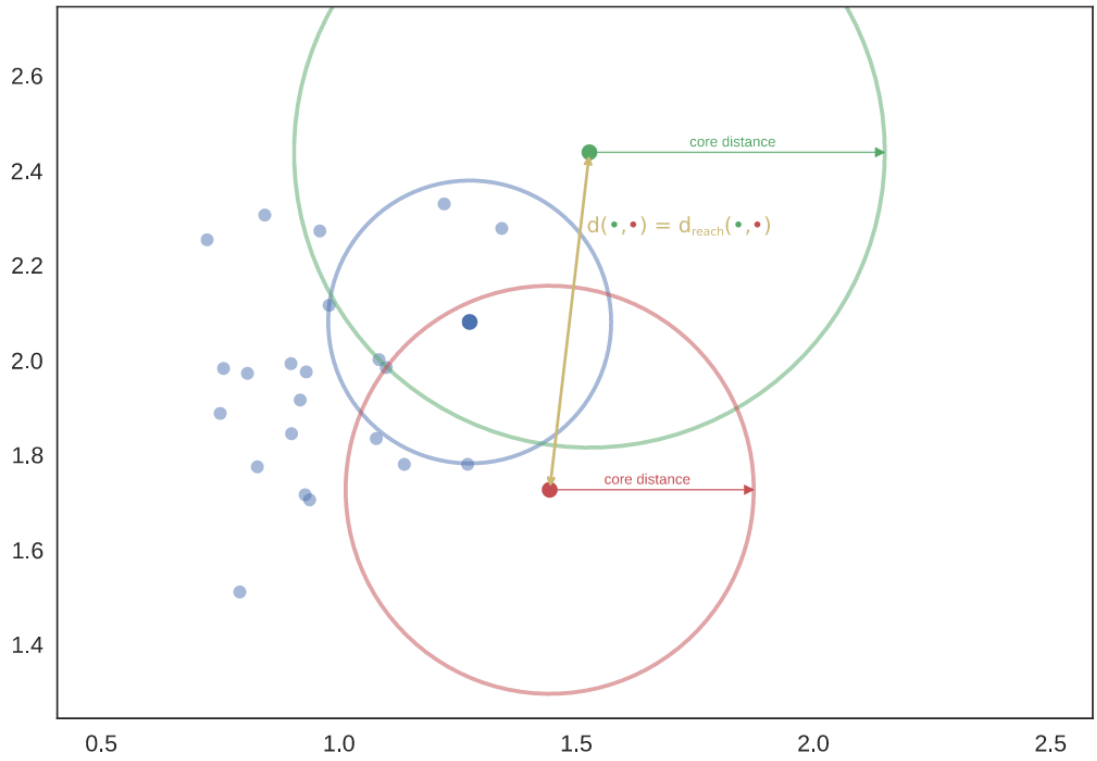


Fig. 1.21: Mutual Reachability and Core Distances between Two Points in a Dataset [16]

Utilizing the mutual reachability distance helps distinguish sparse points from others by at least their core distance, enhancing the clustering's resilience to noise. The dataset is represented as a graph where data objects are vertices connected by weighted edges, with mutual reachability distances serving as the edge weights.

HDBSCAN constructs a minimum spanning tree from this graph and sorts its edges by mutual reachability distance yielding a hierarchical tree structure, or dendrogram. By setting epsilon as a global horizontal cut-off value and selecting clusters with at least mpts (minimum points parameter) points at this density level, we can extract the DBSCAN* clusters corresponding to this epsilon from the hierarchy.

HDBSCAN then focuses on uncovering clusters with varying densities by creating a simplified version of the intricate hierarchical tree, known as the condensed cluster tree. Starting from the root, a cluster split is considered *true* only if both resulting child clusters contain at least mpts (minimum points parameter) objects. If both child clusters have fewer than mpts objects, the cluster is deemed to have vanished at that density level. If only one child cluster has fewer than mpts objects, the parent cluster is seen as having lost some points but remains intact. These "lost" points are treated as noise. This simplification yields a hierarchy of potential clusters at different density levels.

1.3.8.1 HDBSCAN Validation

One of the most challenging aspects of clustering is validation, which involves the objective and quantitative evaluation of clustering results. Various relative validity criteria have been proposed for assessing the quality of globular clusters [19] for example the silhouette score.

The silhouette score is a measure used to evaluate the quality of clustering by calculating how similar an object is to its own cluster compared to other clusters. For each data point i, the silhouette score $s(i)$ is defined as follows:

1. Calculate $a(i)$: The average distance between i and all other points in the same cluster. This measures the cohesion within the cluster.
2. Calculate $b(i)$: The average distance between i and all points in the nearest neighboring cluster. This measures the separation from the nearest cluster.
3. Compute $s(i)$: The silhouette score for point i is given by:

$$s(i) = \frac{b(i) - a(i)}{\max(a(i), b(i))}$$

The silhouette score ranges from -1 to 1:

- 1: The point is well-matched to its own cluster and poorly matched to neighboring clusters.
- 0: The point is on or very close to the boundary between two clusters.
- -1: The point might have been assigned to the wrong cluster.

The overall silhouette score for the clustering is the average silhouette score of all points.

Unfortunately, not all data consist of globular clusters (see Figure 1.22). To address this issue, Density-Based Clustering Validation (DBCV) has been developed.

For each cluster C_i, we start by computing the core distances of the objects within C_i, which are used to determine the Mutual Reachability Distances (MRDs) for all pairs of objects in C_i. Using the MRDs, we construct a Minimum Spanning Tree (MST) for each cluster. This procedure is repeated for all clusters, resulting in l minimum spanning trees, one for each cluster.

Using these MSTs, we introduce a density-based clustering validation index based on the concepts of density sparseness and density separation. Density sparseness of a cluster is the maximum edge weight of its MST, representing the area of lowest density within the cluster. Density separation between two clusters is the minimum MRD between their

objects, representing the densest area between the clusters. These concepts are combined to form the validity index, DBCV.

We define internal edges in the MST as those excluding edges with an endpoint of degree one, and internal objects as those excluding objects with degree one. The density sparseness and separation of clusters are formalized in the following definitions.

Density Sparseness of a Cluster The Density Sparseness of a Cluster C_i is the maximum edge weight of the internal edges in its MST, constructed using core distances of the objects in C_i.

Density Separation The Density Separation between clusters C_i and C_j is the minimum reachability distance between their internal nodes.

We then compute the density-based quality of a cluster as follows.

Validity Index of a Cluster The validity of a cluster C_i is defined as:

$$V_C(C_i) = \frac{\min_{1 \leq j \leq l, j \neq i}(DSPC(C_i, C_j)) - DSC(C_i)}{\max(\min_{1 \leq j \leq l, j \neq i}(DSPC(C_i, C_j)), DSC(C_i))}$$

A positive validity index indicates better density compactness than separation, while a negative index indicates the opposite.

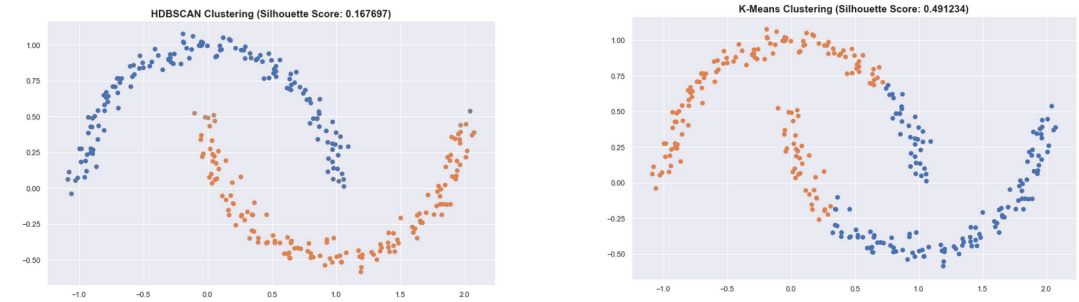
Validity Index of a Clustering The Validity Index of a clustering solution $C = \{C_i\}$, $1 \leq i \leq l$, is the weighted average of the validity indices of all clusters:

$$DBCV(C) = \sum_{i=1}^{l} \frac{|C_i|}{|O|} V_C(C_i)$$

The DBCV index ranges from -1 to +1, with higher values indicating better density-based clustering solutions.

1.3.9 Count Vectorizer

CountVectorizer stands as a foundational tool within natural language processing (NLP), serving to transform a set of textual documents into a matrix reflecting the frequency of tokens. Its operation involves breaking down documents into tokens and tallying the instances of each token, be it a word or character. It establishes a lexicon encompassing all tokens present in the corpus, encoding each document as a sparse matrix. Here, rows



(a) HDBSCAN clustering (Silhouette Score: 0.17) (b) K-Means clustering (Silhouette Score: 0.49)

Fig. 1.22: Limitations of silhouette score as an evaluation metric for HDBSCAN clustering

correspond to documents while columns represent tokens, with each cell denoting the token count within the respective document.

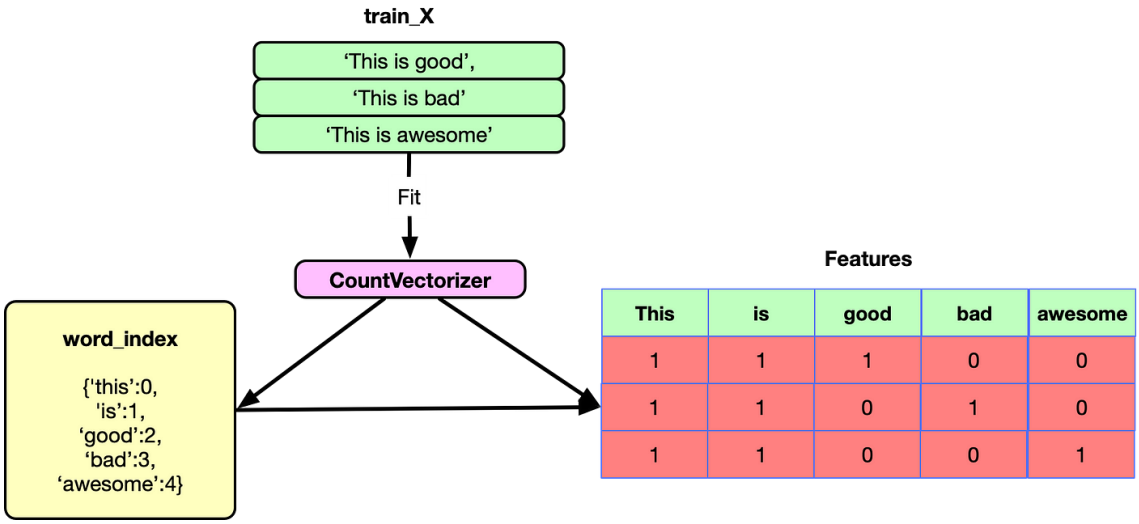


Fig. 1.23: CountVectorizer [22]

In employing CountVectorizer, various parameters come into play. These include considerations such as whether to treat words as case-insensitive, the inclusion or exclusion of stopwords—commonly disregarded in many NLP tasks for their limited utility. Furthermore, options exist to filter out terms that occur excessively or possess minimal document frequency. Users can also opt for n-grams, which capture sequences of tokens beyond individual words. Lastly, there's the option to binarize the counts, setting occurrences of terms to 1 or 0 based on their presence, rather than accounting for frequency. This approach proves valuable when the frequency of a term holds little significance for the machine learning model.

1.3.10 TF-IDF

Term frequency-inverse document frequency (tf-idf) is a measure of the importance of a word in a document. It first calculates the term frequency, which is the frequency of a word in a document. This is determined by dividing the number of occurrences of the

word in the document by the total number of words in that document. Then, it is divided by the document frequency, which is the number of documents in which the word occurs. By employing this approach, tf-idf assigns higher values to words that occur frequently within a document while diminishing the importance of words that are common across many documents. This reflects the intuition that the specificity of a term can be quantified as an inverse function of the number of documents in which it occurs.[23]

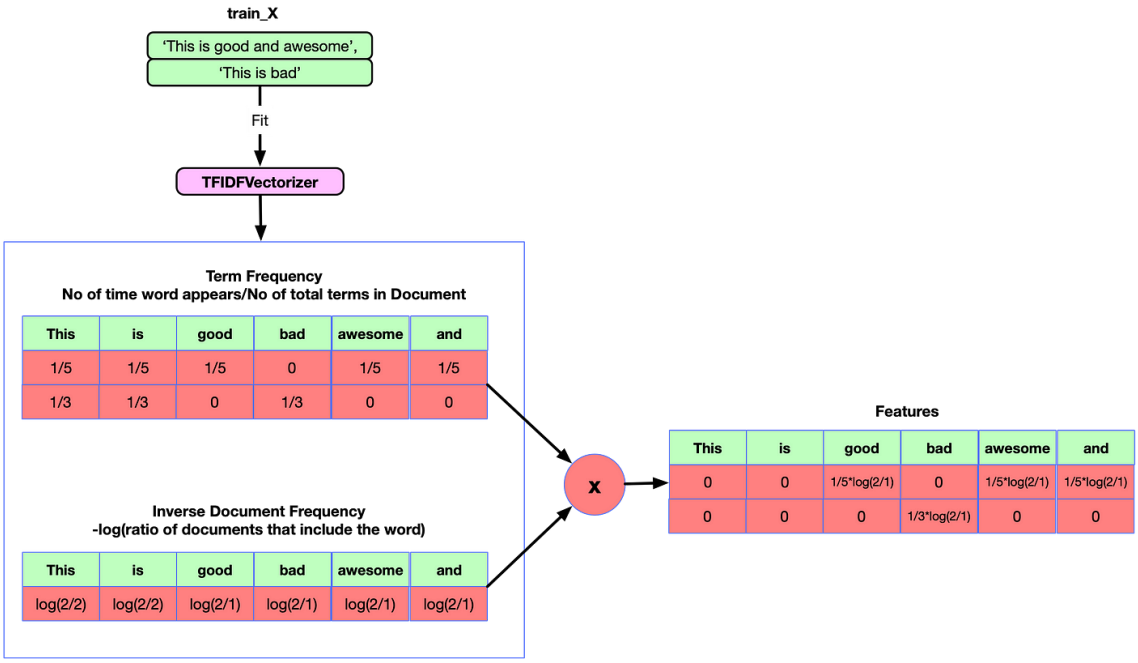


Fig. 1.24: TF-IDF [22]

Mathematically, tf-idf is represented as follows:

$$\text{tf-idf}(t, d, D) = \text{tf}(t, d) \times \text{idf}(t, D)$$

where:

$$\text{tf}(t, d) = \frac{\text{number of occurrences of term } t \text{ in document } d}{\text{total number of words in document } d}$$

and:

$$\text{idf}(t, D) = \log\left(\frac{\text{total number of documents in collection } D + 1}{\text{number of documents containing term } t + 1}\right)$$

tf-idf is usually used in various natural language processing tasks such as information retrieval, text mining, and document classification, where it helps in identifying the significance of words within documents and distinguishing them based on their relevance to specific topics or themes.

Class-specific tf-idf or (ctf-idf) is a modified version of tf-idf where the inverse document frequency (idf) is computed by aggregating all documents within the same class into one virtual document, and then l1 normalized. Similarly, the term frequency is calculated by treating all documents from the same class as one (refer to Figure 1.25).

This modification allows ctf-idf to capture the importance of terms within a specific class while accounting for the frequency of terms across the entire class. By aggregating documents within the same class, ctf-idf emphasizes terms that are distinctive to that class while downplaying terms that are common across multiple classes.

ctf-idf is particularly useful in tasks where class-specific information is important, such as text classification or sentiment analysis, as it helps in identifying and highlighting class-specific features or characteristics within the documents.

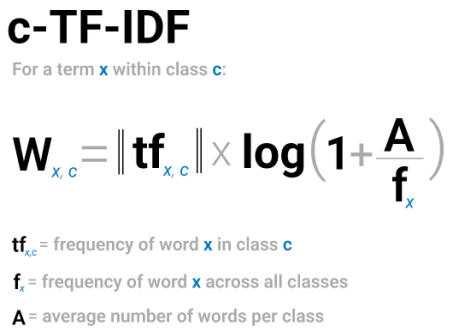


Fig. 1.25: Class-based TF-IDF [13]

1.3.11 MMR

Maximal Marginal Relevance (MMR) is a method used in information retrieval and summarization tasks to select a diverse set of items from a larger pool based on their relevance to a query or topic. It aims to balance between relevance and diversity, ensuring that the selected items are both highly relevant to the query and diverse from each other. The intuition behind MMR is to maximize the marginal relevance of each selected item while minimizing redundancy among them. In practice, MMR achieves this by iteratively selecting items that have high relevance to the query while also considering their dissimilarity to previously selected items, thus promoting diversity in the final selection.

Given a set of candidate items C, a query or topic q, and a parameter λ that controls the trade-off between relevance and diversity, MMR selects items based on the following formula for each candidate item $c_i \in C$:

$$MMR(c_i) = \lambda \times \text{similarity}(c_i, q) - (1 - \lambda) \times \max_{c_j \in R} \text{similarity}(c_i, c_j)$$

Where:

- similarity(c_i, q) measures the relevance of item c_i to the query q.

- similarity(c_i, c_j) measures the similarity between item c_i and previously selected items c_j in the selected set R.
- λ is a parameter that controls the trade-off between relevance and diversity. A higher value of λ emphasizes relevance more, while a lower value promotes diversity.
- R is the set of previously selected items.

The goal of MMR is to select items c_i that maximize the marginal relevance score $MMR(c_i)$, where high relevance to the query and low similarity to previously selected items are favored. By iteratively selecting items based on this criterion, MMR constructs a diverse set of items that are both relevant to the query and distinct from each other, making it an effective method for information retrieval and summarization tasks.

1.4 Natural Language Processing

Natural Language Processing (NLP) is an interdisciplinary field that merges linguistics with computer science, aiming to equip computers with the ability to comprehend and manipulate human language. This involves the application of various techniques, including rule-based and probabilistic methods such as statistical and neural network-based approaches, to analyze natural language data such as text or speech corpora. The ultimate goal is to develop computer systems capable of comprehending the nuanced linguistic nuances present in documents, thereby enabling accurate classification, organization, and extraction of insights and information from them. One notable application of NLP is topic modeling.

1.4.1 Word Representation

Word representation is a fundamental concept in NLP. It involves the process of transforming words or phrases from human-readable text into numerical vectors that machine learning algorithms can understand and process efficiently.

The traditional approach to representing words in NLP relied on techniques such as one-hot encoding, where each word is represented by a high-dimensional binary vector, with a 1 at the index corresponding to the word's position in a predefined vocabulary and 0s elsewhere. However, one-hot encoding suffers from several limitations, including high dimensionality and lack of semantic information (refer to Figure 1.26).

Word representation techniques aim to address these limitations by capturing semantic and syntactic similarities between words, enabling algorithms to understand relationships and meanings encoded in the text. One of the most popular word representation techniques is word embeddings (see Figure1.27).

Word embeddings are dense, low-dimensional vectors that represent words in a continuous vector space, where similar words are closer to each other in the space. These embeddings are learned from large text corpora using unsupervised learning algorithms

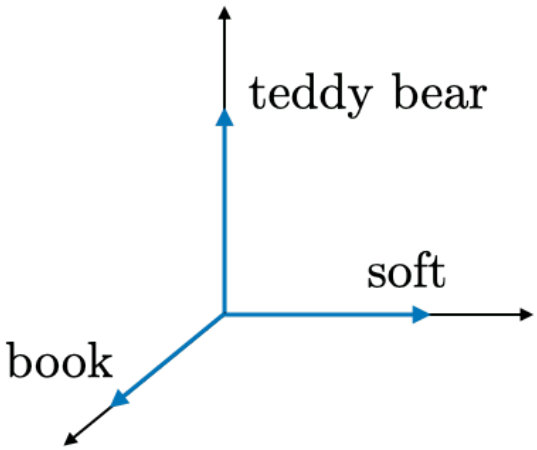


Fig. 1.26: One-hot Representation [5]

such as Word2Vec, GloVe, or FastText. In these algorithms, words are mapped to vectors in a way that preserves semantic relationships based on their co-occurrence patterns in the corpus.

Word embeddings have several advantages over traditional one-hot encoding, including:

- Semantic Similarity: Word embeddings capture semantic similarities between words, allowing algorithms to understand relationships such as synonymy, antonymy, and analogies.
- Dimensionality Reduction: Word embeddings are low-dimensional compared to one-hot encoding, which reduces the computational complexity of NLP tasks and enables better generalization.
- Transfer Learning: Pre-trained word embeddings can be transferred to downstream NLP tasks with limited labeled data, facilitating faster and more efficient model training.
- Contextual Information: Word embeddings can capture contextual information, as the meaning of a word can vary depending on its surrounding words in a sentence. Contextual word embeddings, such as those produced by models like BERT and GPT, take into account the entire context of a word within a sentence or document.

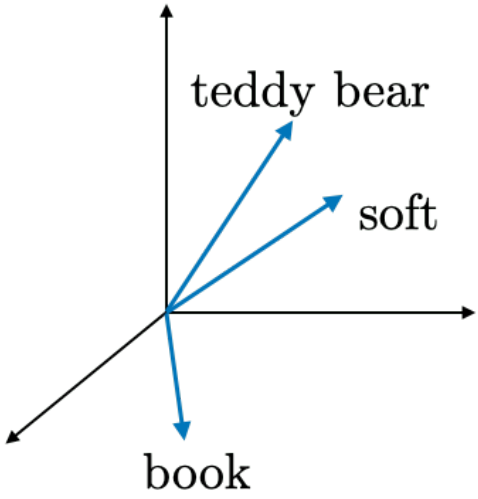


Fig. 1.27: Word Embedding [5]

1.4.2 Similarity Metrics

1.4.2.1 Cosine similarity

Cosine similarity is a metric used to measure the similarity between two vectors by calculating the cosine of the angle between them. In the context of word embeddings, each word is represented as a high-dimensional vector, and cosine similarity is employed to quantify the degree of similarity between these vectors.

Mathematically, the cosine similarity between two vectors **a** and **b** is defined as:

$$\text{similarity}(\mathbf{x}, \mathbf{y}) = \frac{\mathbf{x} \cdot \mathbf{y}}{\|\mathbf{x}\|\|\mathbf{y}\|} = cos(\theta)$$

Where:

- $\mathbf{x} \cdot \mathbf{y}$ denotes the dot product of vectors $\mathbf{x}$ and $\mathbf{y}$.
- $\|\mathbf{x}\|$ and $\|\mathbf{y}\|$ represent the Euclidean norms of vectors $\mathbf{x}$ and $\mathbf{y}$, respectively.
- θ represents the angle between the vectors $\mathbf{x}$ and $\mathbf{y}$.

Cosine similarity yields a value between -1 and 1. A value of 1 indicates that the vectors are identical, 0 denotes orthogonality (i.e., no similarity), and -1 implies complete dissimilarity.

In the context of word embeddings, cosine similarity provides a measure of semantic similarity between words. Words with similar meanings will have vectors that point in

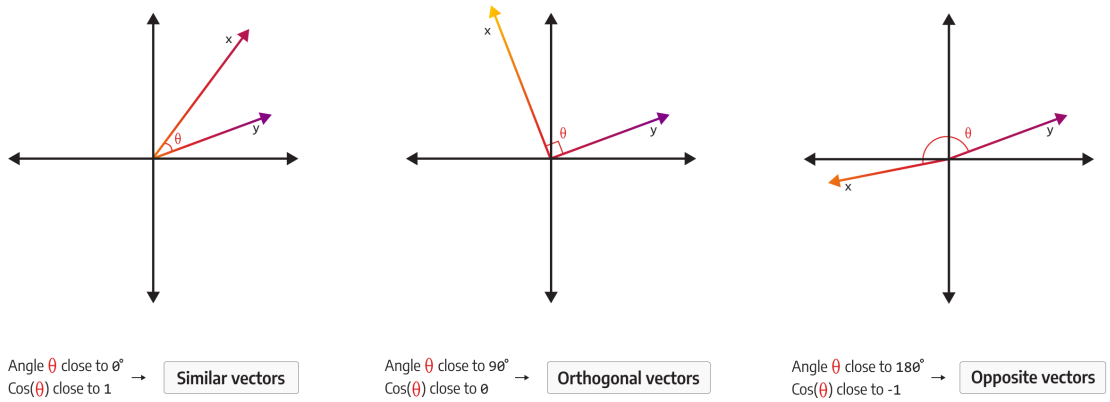


Fig. 1.28: Representation of the cosine similarity [24]

similar directions in the embedding space, resulting in a higher cosine similarity score. Conversely, words with dissimilar meanings will have vectors pointing in different directions, leading to lower cosine similarity scores.

Cosine similarity is extensively used in NLP tasks such as document similarity, query expansion, and clustering. It allows algorithms to understand the semantic relationships between words and documents, thereby enabling more accurate and contextually relevant analysis of textual data.

1.4.2.2 Euclidian Distance

While cosine similarity checks for the similarity of two vectors by assessing how aligned they are, Euclidean distance takes into consideration the intensity of each vector as well. The larger the Euclidean distance between two vectors, the less similar they are. Mathematically, the Euclidean distance between two vectors $u = (u_i)_i$ and $v = (v_i)_i$ is defined as follows:

$$d(u, v) = \sqrt{\sum_i (u_i - v_i)^2}$$

1.4.2.3 Hamming Distance

Hamming distance, like Euclidean distance, is a measure of dissimilarity. It calculates the difference between two strings or vectors of equal length by counting the number of dissimilarities between them.

Mathematically, the Hamming distance between two strings or vectors u and v of equal length n is defined as:

$$d_H(u, v) = \sum_{i=1}^{n} f(u_i - v_i)$$

where $f(0) = 0$ and $f(x) = 1$ if $x \neq 0$. For two binary vectors (vectors comprising solely of 0s and 1s), the Hamming distance simplifies to:

$$d_H(u, v) = \sum_{i=1}^{n} |u_i - v_i|$$

There are some variants of the Hamming distance which divide the sum by the length of the vectors n, but as a measure of dissimilarity, both are equivalent.

1.4.3 Quantization

Quantization is a fundamental technique used in signal processing and data compression to reduce the precision of numerical data. Its primary purpose is to represent large sets of data with a smaller range of values, thus saving storage space and potentially speeding up computation.

Quantization is employed in various fields such as image and audio compression, where reducing the number of bits used to represent each sample can significantly decrease file size while maintaining an acceptable level of perceptual quality. Recently, it has also been introduced in embeddings. State-of-the-art models produce embeddings with 1024 dimensions, each encoded in float32 or float64, requiring 4 or 8 bytes per dimension. Consequently, performing retrieval over 250 million vectors demands approximately 1TB or even 2TB of memory!

During quantization, each original data value x is mapped to a discrete value within a finite set of possible values. This mapping is typically done by dividing the range of the original data into intervals and assigning each interval a representative value. By doing so, quantization discards some of the finer details of the original data, resulting in a loss of information.

Quantization is inherently a lossy process because the reduced precision means that the reconstructed data will not be identical to the original. This loss of information can lead to perceptible degradation in quality, especially in applications where high precision is essential, such as medical imaging or scientific data analysis.

In scalar quantization, each data point x_i is independently mapped to a discrete value. This is achieved by dividing the range of the data into intervals and assigning a representative value (quantization level) to each interval. Scalar quantization is widely used in various compression algorithms due to its simplicity and efficiency. The mapping function can be expressed as:

$$Q(x_i) = \text{round}\left(\frac{x_i}{\Delta}\right) \cdot \Delta$$

where Δ is the quantization step size.

Binary quantization is a special case where each data point is mapped to one of two possible values (usually 0 or 1). This form of quantization is particularly efficient in terms of storage space and computational complexity, making it suitable for applications where resources are limited, such as embedded systems or neural network weight quantization. To quantize embeddings to binary, we simply threshold normalized embeddings at 0:

$$f(x_i) = \begin{cases} 0 & \text{if } x_i \leq 0 \\ 1 & \text{if } x_i > 0 \end{cases}$$

Interestingly, while binary quantization is more lossy compared to scalar quantization, some models ("all-MiniLM-L6-v2" for example) exhibit stronger performance when trained or deployed with binary quantized weights. This phenomenon can be attributed to various factors, including the ability of binary quantization to introduce noise that acts as a form of regularization, preventing overfitting and improving generalization performance. Additionally, binary quantization can facilitate faster inference and reduced memory requirements, making it appealing for real-time applications or deployment on resource-constrained devices.

1.4.4 Topic Modeling

Topic modeling is a technique used in natural language processing and machine learning to discover hidden thematic structures within a collection of documents. It aims to uncover the underlying topics or themes that are present in the text data, without any prior knowledge of these topics. The primary goal of topic modeling is to automatically identify and extract meaningful patterns of words that co-occur frequently across documents, thus providing insights into the main themes or subjects discussed in the corpus.

At its core, topic modeling assumes that each document in the corpus is a mixture of several topics, and each topic is characterized by a distribution of words. By analyzing these word distributions across documents, topic modeling algorithms can infer the underlying topics and their prevalence in the dataset. One of the most popular techniques for topic modeling is Latent Dirichlet Allocation (LDA), which models documents as mixtures of topics and topics as mixtures of words.

Topic modeling finds applications in various domains such as text summarization, information retrieval, document clustering, and recommendation systems. It enables researchers and analysts to gain insights into large text collections, identify trends, and explore patterns within the data without the need for manual annotation or supervision.

1.4.4.1 BERTopic

BERTopic is a topic modeling technique that leverages the transformers' architecture and c-TF-IDF to create dense clusters allowing for easily interpretable topics whilst keeping important words in the topic descriptions.

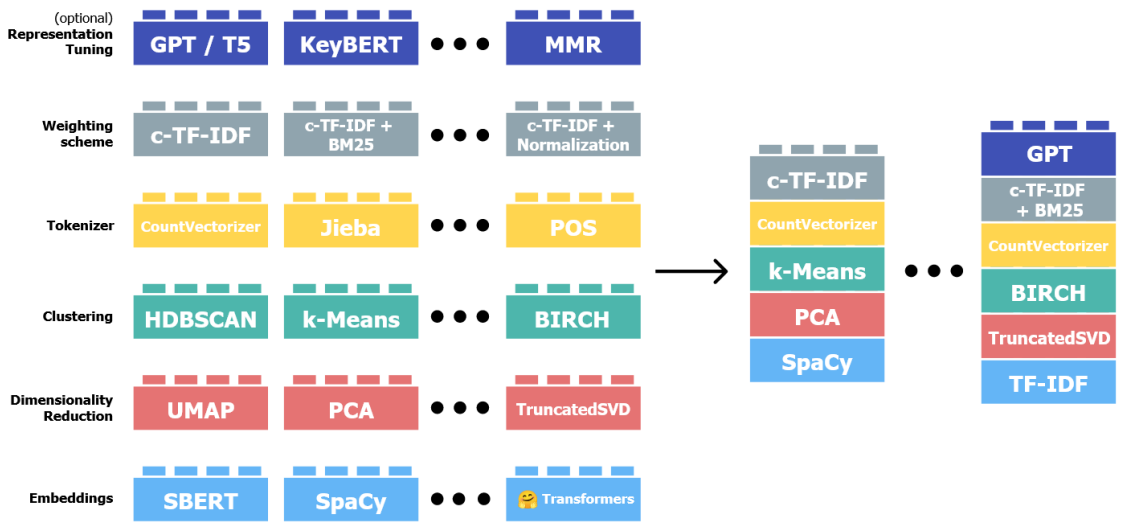


Fig. 1.29: BERTopic Algorithm's modularity [12]

Embeddings In BERTopic, the initial step involves converting documents into numerical representations, a process crucial for various natural language processing tasks. While numerous methods exist for this purpose, BERTopic defaults to sentence-transformers, which excel in semantic similarity optimization. These models efficiently create embeddings for either documents or sentences, facilitating tasks like clustering.

Embeddings serve as numerical representations of complex objects, such as text, images, or audio, condensed into n-dimensional vectors. Once objects are transformed, their similarity can be determined by comparing the embeddings. This functionality underpins numerous applications, including recommendation systems, retrieval, one-shot or few-shot learning, outlier detection, and paraphrase detection, among others.

Within BERTopic, users have the flexibility to select from a range of sentence-transformers models. However, two models are predefined as defaults: "all-MiniLM-L6-v2" and "paraphrase-multilingual-MiniLM-L12-v2". These default options offer optimized performance for diverse clustering tasks.

Dimensionality Reduction Given our aim to establish various topic clusters, we'll frequently encounter several challenges due to the high dimensionality of transformer embeddings, including:

- Data sparsity: As the data becomes sparse, much of the high-dimensional space remains unoccupied, posing difficulties for clustering and classification tasks.
- Increased computation: Higher dimensions necessitate more computational resources and time for data processing.
- Overfitting: Models with higher dimensions may become excessively complex, fitting noise rather than the underlying pattern, thereby reducing generalization to new data.

- Performance degradation: Algorithms relying on distance measurements, such as k-nearest neighbors, may experience a decline in performance.

Collectively, these challenges are referred to as the curse of dimensionality. One solution is to reduce the dimensionality of embeddings to a manageable level (e.g., 5) for clustering algorithms.

BERTopic defaults to using UMAP because it effectively captures both local and global high-dimensional spaces in lower dimensions. However, alternatives like PCA are also available for users to explore. Since BERTopic assumes some level of independence between steps (refer to Figure 1.29), other dimensionality reduction algorithms can be employed as well.

Clustering After transforming documents into numerical representations using techniques such as sentence-transformers within BERTopic, and subsequently reducing the dimensions of the vectors representing these documents, the subsequent phase usually entails clustering.

Within BERTopic, clustering endeavors to categorize documents into clusters or groups grounded on their semantic similarity, as reflected in the condensed embeddings. The objective of clustering is to reveal latent patterns or themes inherent in the document set. Through the aggregation of akin documents, clustering aids in discerning prevalent topics, themes, or categories embedded within the corpus.

BERTopic defaults to using HDBSCAN for this clustering step. However, similar to the other steps, users have the flexibility to employ any clustering algorithm that suits their specific needs.

The outcome of this process is an array of unlabeled clusters, each containing multiple documents.

Topic Representation Now, the subsequent step involves topic representation. We analyze the term frequency within each cluster and then utilize ctf-idf or an alternative variant as a weighting scheme. This allows us to construct a word representation for each cluster, providing insight into the thematic focus of each (refer to Figure 1.30).

Optionally, we may incorporate a representation tuning step to eliminate redundant words, thereby enhancing diversity within the representation. This, in turn, contributes to a more robust and meaningful depiction of each cluster's topic.

1.4.5 Sentiment Analysis

Sentiment analysis is a natural language processing task that involves determining the sentiment or opinion expressed in a piece of text. The purpose of sentiment analysis is to automatically classify text into categories such as positive, negative, or neutral sentiment,

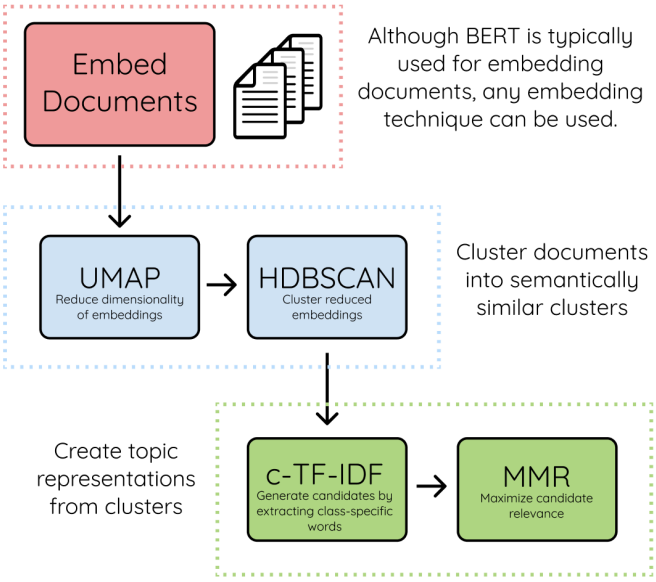


Fig. 1.30: BERTopic Algorithm [10]

providing valuable insights into the opinions, attitudes, and emotions conveyed in textual data.

This analysis is particularly useful in various applications, including:

- Social Media Monitoring: Analyzing sentiment in social media posts, reviews, or comments to gauge public opinion about products, services, events, or brands.
- Customer Feedback Analysis: Understanding customer satisfaction levels, identifying areas for improvement, and addressing customer concerns based on feedback analysis.
- Market Research: Evaluating consumer sentiment towards products or trends, identifying emerging issues or trends, and making informed business decisions.

Sentiment analysis can complement topic modeling by providing additional insights into the emotional context surrounding identified topics or themes. While topic modeling identifies the main themes or subjects discussed in a corpus, sentiment analysis helps understand the sentiment associated with each topic, providing a more comprehensive understanding of the underlying attitudes and opinions expressed by users.

Sentiment analysis is typically performed using machine learning techniques, where a classifier is trained on labeled datasets to predict the sentiment of unseen text. Classification models, especially those based on deep learning architectures like transformers, have shown excellent performance in sentiment analysis tasks.

Utilizing the capabilities of transformers' embeddings, which may be trained on extensive textual data, and classification models such as standard feedforward neural networks, sentiment analysis can be executed with efficiency and precision. This approach offers valuable insights into the sentiments conveyed within textual data.

Chapter 2

Experimental analysis and Discussion

In this section, we will outline the methodology employed for data collection, offering both a succinct descriptive overview and an in-depth analysis of the outcomes at each stage of the topic modeling process using BERTopic. Following this, we will present the final results and provide an interpretation and discussion regarding their quality. We'll also delve into the sentiment analysis findings, tracing the data's progression through each preprocessing stage and explaining its culmination. Furthermore, we will examine the significance of these results, emphasizing their strengths and weaknesses and discussing their implications.

2.1 Data Collection

In our data collection process, we focused on analyzing ESG (Environmental, Social, and Governance) data sourced from English-language press media platforms, given their prevalence in the regions of interest.

Our data gathering targeted the most populous African countries, as they are more likely to have press outlets publishing content in English. Specifically, we focused on the top 20 African nations by population, which together account for approximately 81.8% of Africa's total population [28]. These countries are Nigeria, Ethiopia, Egypt, DR Congo, Tanzania, South Africa, Kenya, Uganda, Sudan, Algeria, Morocco, Angola, Ghana, Mozambique, Madagascar, Ivory Coast, Cameroon, Niger, Mali, and Burkina Faso.

Our methodology is built upon a framework (see Table 2.1) that correlates ESG topics with the Sustainable Development Goals (SDGs) set by the United Nations. This framework guides our analysis, linking ESG issues with broader global sustainability goals. To enrich this framework, we incorporated relevant keywords associated with ESG topics. Initially, we sourced keywords from research papers on reputable platforms.

The first phase of our data collection involved determining our search queries. We aimed to represent Africa adequately in our queries, focusing on the aforementioned 20 countries. We also included a query for "Africa," making a total of 21 search terms.

For keyword identification, we used the "IMPACT" column from the c40 table (see Table 2.1) as our baseline. Through the Arxiv API, we accessed abstracts from the 200 most relevant research papers for each keyword. We then used keyBERT to extract relevant keywords from these abstracts, retaining only those that appeared at least five times across the 200 abstracts for each "IMPACT" category.

We extended this process to the MDPI platform via Bing research, adding an ESG lexicon [3] to the original keywords from the "IMPACT" column. After removing duplicates

and converting keywords to lowercase, we had a set of 10,941 queries (see Figure 2.1).

THEME	**IMPACT GROUP**	**IMPACT**	**SDG #**
Social	Health	Physical health	2, 3
Social	Health	Mental health	3
Social	Quality of life and urban liveability	Housing	7, 11
Social	Quality of life and urban liveability	Work-life balance	1, 5, 8, 11
Social	Quality of life and urban liveability	Peace and security	5, 10, 11, 16
Social	Quality of life and urban liveability	Attractiveness	11
Social	Culture	Cultural richness and heritage	4, 11
Social	Culture	Education	4, 8, 12
Social	Culture	Environmental and health awareness and behaviour	5
Social	Culture	Social participation	11
Social	Institutions	Good governance	11, 12, 14, 16, 17
Economics	Wealth and economy	Environmental and health awareness and behaviour	1, 8, 17
Economics	Wealth and economy	Employment	1, 8, 9, 11
Economics	Wealth and economy	Economic innovation, dynamism and competitiveness	8, 9
Economics	Wealth and economy	Private wealth	10
Economics	Wealth and economy	Public budget	17
Economics	Wealth and economy	Sustainable production and consumption	2, 6, 8, 9, 11, 12, 13
Environmental	Environmental quality	Biodiversity	14, 15, 11
Environmental	Environmental quality	Climate Change	7, 11, 13
Environmental	Environmental quality	Air quality	3, 11, 15
Environmental	Environmental quality	Noise	No related SDG
Environmental	Environmental quality	Soil quality	3, 11, 12, 15
Environmental	Environmental quality	Light pollution	No related SDG
Environmental	Environmental quality	Water quality	3, 6, 11, 14
Environmental	Environmental quality	Temperature	13

Table 2.1: ESG and SDGs Framework (c40.org)

Using Bing for our searches provided links, authors, titles, snippets, and publication dates, but not the full articles. To access the articles, we cleaned the data by removing empty and duplicate rows. We then categorized the links into MSN, Yahoo, and others. MSN and Yahoo were chosen for their comprehensive news coverage from various curated sources. We scraped articles from MSN and Yahoo using their respective APIs and used Trafilatura [7] for the remaining links.

From our 10,941 queries, we obtained 29,329 unique links: 7,463 from MSN, 721 from Yahoo, and 21,145 from other sources. After filtering out invalid URLs and deleted articles, we had 6,787 unique MSN articles, 655 from Yahoo, and 19,153 from other sources, totaling

	query
0	"flows"+"Africa"
1	"flows"+"Nigeria"
2	"flows"+"Ethiopia"
3	"flows"+"Egypt"
4	"flows"+"DR Congo"
...	...
10936	"occupational"+"stress"+"Ivory Coast"
10937	"occupational"+"stress"+"Cameroon"
10938	"occupational"+"stress"+"Niger"
10939	"occupational"+"stress"+"Mali"
10940	"occupational"+"stress"+"Burkina Faso"

10941 rows × 1 columns

Fig. 2.1: The Query DataFrame

26,595 unique articles.

During preprocessing, we found formatting issues with articles from "citizen.co.za." To remedy this, we developed a scraper for the website and retrieved 483 properly formatted articles to replace the problematic ones.

2.2 Data Preprocessing

The preprocessing step is indeed critical in every NLP task. It serves as the foundation that prepares the data for the modeling phase, ensuring that the text is in a format that can be effectively analyzed and interpreted by machine learning algorithms. Proper preprocessing can significantly impact the performance and accuracy of the models, as it addresses various challenges like noise, inconsistencies, and irrelevant information in the text data. Techniques commonly used in preprocessing include tokenization, stemming or lemmatization, removing stop words, handling special characters, and converting text to lowercase, among others. By carefully handling these aspects during preprocessing, we can enhance the quality of the data and improve the overall results of the NLP task.

During the preprocessing phase, several steps were undertaken to refine and prepare the scraped data for subsequent analysis. The following steps were followed:

- Text Length Filtering: Texts larger than 4000 tokens and those smaller than 5 words were removed to focus on content of suitable length.
- Language Filtering: Only English language data was retained for further processing to ensure consistency and ease of analysis.

- URL and Phone Number Removal: Sentences containing URLs and phone numbers were systematically eliminated from the text to maintain focus on meaningful content and avoid noise.
- Special Word Elimination: Sentences containing specific words, indicative of advertisements or low relevance, were identified and removed to improve data quality.
- Numeric and Special Character Removal: Numbers and special characters were stripped from the text as they contribute minimally to contextual understanding and may interfere with subsequent processing steps.
- Text Segmentation into Paragraphs: Texts were segmented into paragraphs to preserve contextual coherence, as paragraphs typically encapsulate distinct topics or ideas in press media.
- Short Paragraph Elimination: Paragraphs containing four words or less were discarded, as they typically lack substantive content and provide minimal value for analysis.
- Non-Unique Paragraph Filtering: Non-unique paragraphs were examined to identify repeated content across different sources. While retaining non-unique paragraphs indicative of significant topics, signatures, or ads were excluded.
- Error Message Verification: Verification was conducted to ensure the presence of error messages where applicable, maintaining data integrity and completeness.
- Long Paragraph Analysis: Long paragraphs were inspected to explore possibilities for formatting into smaller, more manageable segments. This step primarily served as a visualization exercise to aid in subsequent analysis.

Since we plan to filter out topics that aren't relevant to ESG during the topic modeling step, minor noise or inconsistencies in the data during preprocessing may not significantly impact the final topic modeling results. However, it's still important to strike a balance between thorough cleaning and efficiency, as overly aggressive preprocessing could potentially remove valuable information or context that might be beneficial for the modeling process.

While the noise might not directly affect the relevance of the topics, it's essential to ensure that the data remains coherent enough for the models to extract meaningful patterns and insights related to ESG topics. Therefore, a moderate level of preprocessing to address major inconsistencies or irrelevant content while preserving the integrity and context of the data is generally advisable. This approach can help streamline the modeling process and yield more accurate and interpretable results.

2.3 Topic Modeling

We are employing BERTopic as our topic modeling framework. It combines various components, including the embedding model, dimensionality reduction, clustering, weighting scheme, representation tuning, and vectorization, into an integrated topic modeling

pipeline. Due to its modular design, we are adopting an iterative approach to address any challenges we encounter during the process.

2.3.1 Baseline Model

In our initial model, we had not yet completed our data collection. Without the publication dates of many articles, which later proved crucial for analyzing sentiment trends, our insights were limited. The preprocessing step progressed concurrently with model development, meaning we had not yet executed all the preprocessing steps mentioned earlier. Our dataset was structured at the sentence level, comprising a total of 1.2 million sentences. For this model, we used the first 400,000 sentences (see Table 2.2).

	verbatim
0	On his return from the US Zuma apparently told family members someone close to him had made an attempt on his life
1	Because South Africa uses coal to generate electricity this added pressure to the power grid means electric cars wont really make much of a difference in reducing ones carbon footprint if driven in South Africa
2	The third shift which will be implemented from August 2019 is creating an additional 1 200 jobs at the Pretoria based plant which will take the total number of South Africans employed by Ford Motor Company of Southern Africa to 5 500
3	It is no secret that we are not being paid well as teachers despite challenges we face every day in schools
4	to support school infrastructure development and ensure safe learning in schools
5	We are saying let the nation which includes teachers see education as an essential service so that whatever we do we protect education
6	South Africa is not being effective with its waste management particularly in its use with plastic
7	Doctors at Charlotte Maxeke Johannesburg Academic Hospital one of South Africas foremost public health institutions are being forced to risk patients lives
8	We are told that three hospitals are prepared for such an emergency Charlotte Maxeke Academic Hospital Tembisa Hospital and Steve Biko Academic Hospital
9	I can safely say waste is big money and big investment people should never let recycling opportunities go to waste Plastic gold Mpact Felixton recycled 500 000 tons of used and discarded paper and plastic last year
10	The countrys waste management system is struggling to deal with national plastic waste generation with a significant amount of plastic leaking into the environment
...	...
400000	Modise made reference to the July 2021 unrest and looting in Gauteng and KwaZuluNatal when an estimated 15000 members of the military were successfully deployed to quell the riots

Table 2.2: Snippet from data used for the Baseline Model - First 400,000 sentences

So to establish our baseline model, we employed minimally processed data without timestamps and applied the default parameters from the BERTopic algorithm.

We employed the "all-MiniLM-L6-v2" sentence-transformers [21] model, designed for encoding sentences and short paragraphs. This model takes an input text and produces a 384 dimensional dense vector that encapsulates its semantic meaning. This sentence vector can be applied to tasks such as information retrieval, clustering, or measuring sentence similarity. By default, input texts longer than 256 word pieces are truncated. The model was trained on over 1 billion sentence pairs.

For dimensionality reduction, we used UMAP as it's the default choice in our framework. HDBSCAN was selected for clustering, and we employed cTF-IDF as our weighting scheme, built upon the CountVectorizer. Lastly, for representation tuning, we used KeyBERT.

Fig. 2.2: Topic Word Scores for the top 8 topics of the Baseline Model

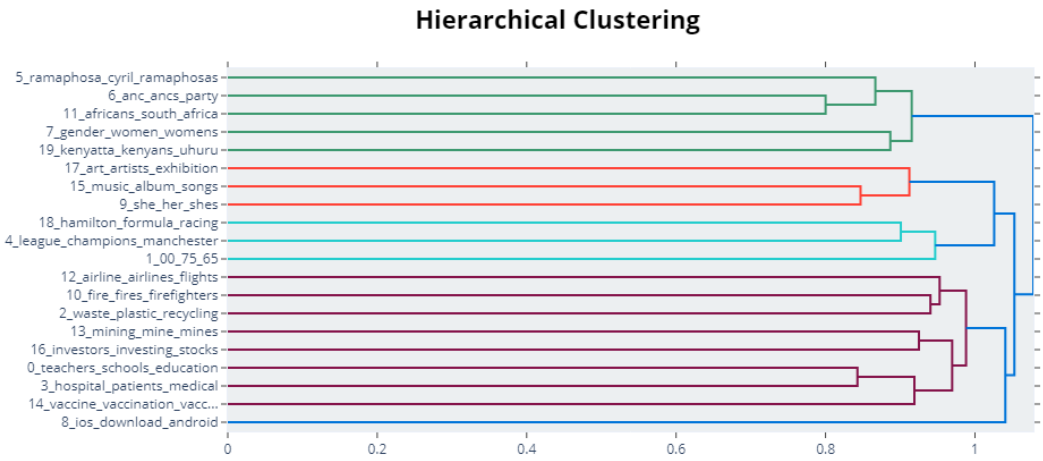


Fig. 2.3: Hierarchical clustering of the top 20 topics of the Baseline Model

From the topic word scores (see Figure 2.2), we can already identify the topics visualized by the model. For instance, Topic 0 is predominantly represented by words like "teachers, schools, education, and students," suggesting a discussion about the education system or school life. Topic 2 features words such as "waste, plastic, recycling, bags," pointing towards a topic related to recycling. Similarly, Topic 3 includes words like "hospital, patients, medical, healthcare, health," indicating a focus on the health and medical field.

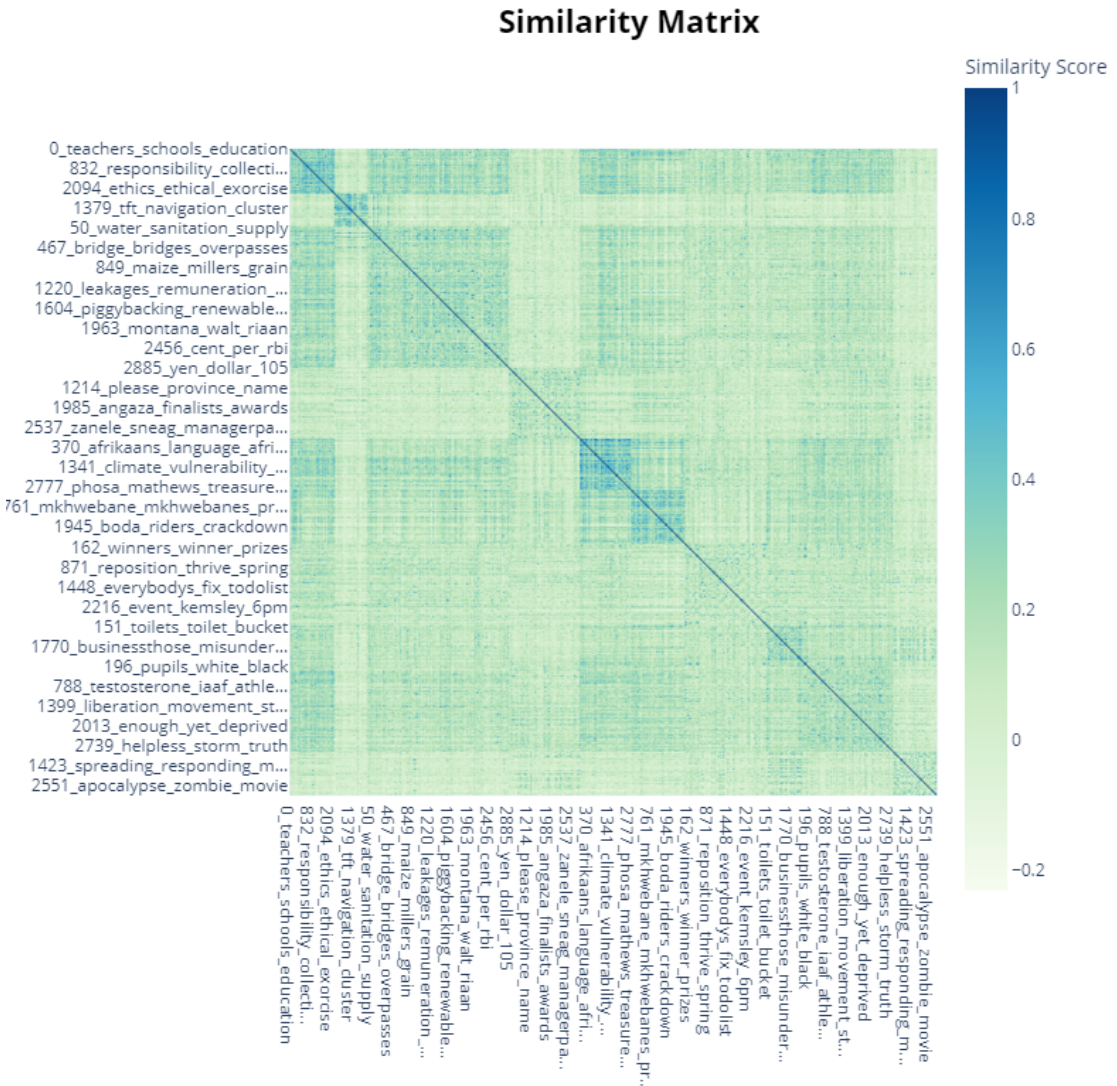


Fig. 2.4: Similarity Matrix for the topics

However, we also observed issues with topic representation in the topic word scores. These issues could arise from the representation model, the preprocessing step, or both. In our analysis, Topic 1 is represented by numbers like 00, 75, 65, 25, which should have been removed during preprocessing as they contribute little to no contextual value and introduce significant noise, particularly for the second-largest topic in terms of document count. Additionally, we noticed redundancy in the representation, such as "plastic" and "plastics" or "vaccine" and "vaccination." This redundancy stems from the use of KeyBERT,

which prioritizes the importance of words to the documents but doesn't penalize similar words within the topic representation.

Topic	Count	Representation	Representative_Docs
-1	207940	['he', 'was', 'she', 'said', 'his', 'her', 'were', 'had', 'they', 'it']	['On his return from the US Zuma apparently told family members someone close to him had made an attempt on his life', 'Because South Africa uses coal to generate electricity this added pressure to the power grid means electric cars wont really make much of a difference in reducing ones carbon footprint if driven in South Africa', 'The third shift which will be implemented from August 2019 is creating an additional 1 200 jobs at the Pretoria based plant which will take the total number of South Africans employed by Ford Motor Company of Southern Africa to 5 500']
0	2350	['teachers', 'schools', 'education', 'school', 'students', 'universities', 'pupils', 'teacher', 'learners', 'student']	['It is no secret that we are not being paid well as teachers despite challenges we face every day in schools', 'to support school infrastructure development and ensure safe learning in schools', 'We are saying let the nation which includes teachers see education as an essential service so that whatever we do we protect education']
1	2252	['00', '75', '65', '25', 'respectively', '50', 'compared', '38', '13', '24']	['00 7', '00 11', '00 25']
2	1909	['waste', 'plastic', 'recycling', 'plastics', 'bags', 'landfill', 'recycled', 'packaging', 'dumping', 'litter']	['I can safely say waste is big money and big investment people should never let recycling opportunities go to waste Plastic gold Mpact Felixton recycled 500 000 tons of used and discarded paper and plastic last year', 'The countrys waste management system is struggling to deal with national plastic waste generation with a significant amount of plastic leaking into the environment', 'South Africa is not being effective with its waste management particularly in its use with plastic']

3	1873	['hospital', 'patients', 'medical', 'healthcare', 'health', 'hospitals', 'doctors', 'nhi', 'care', 'clinic']	['Doctors at Charlotte Maxeke Johannesburg Academic Hospital one of South Africas foremost public health institutions are being forced to risk patients lives', 'We are told that three hospitals are prepared for such an emergency Charlotte Maxeke Academic Hospital Tembisa Hospital and Steve Biko Academic Hospital', 'The hospital has been turning away patients']
4	1854	['league', 'champions', 'manchester', 'liverpool', 'chelsea', 'madrid', 'goal', 'bayern', 'barcelona', 'scored']	['The Premier Leagues longest season is set for a dramatic finish on Sunday as Chelsea Manchester United and Leicester battle for Champions League places with a desperate scrap to avoid relegation at the bottom', 'Still with a shot at winning La Liga Madrid know defeat by Barcelona next weekend would likely end their challenge with Barca currently two ahead of them and Atletico a further four clear with 10 games left to play', 'Manchester Uniteds bid to win the Premier League for the first time in eight years will face a major test when the leaders face champions Liverpool on Sunday']
5	1812	['ramaphosa', 'cyril', 'ramaphosas', 'president', 'phala', 'anc', 'address', 'nation', 'deputy', 'promised']	['It is not in his nature said Ramaphosa', 'The reports were submitted to President Cyril Ramaphosa on 25 November last year', 'This was said by President Cyril Ramaphosa on Monday morning']
6	1672	['anc', 'ancs', 'party', 'conference', 'cosatu', 'nec', 'partys', 'da', 'mangaung', 'leadership']	['So if we are not united as the ANC as society we will not to be able to do it', 'But the members of the commission have been around for some time many of them members of the ANC', 'The greatest enemy of the ANC is the ANC itself and the only thing to save the ANC is the ANC itself']
7	1526	['gender', 'women', 'womens', 'female', 'equality', 'men', 'representation', 'male', 'parity', 'girls']	['But there is still much work to be done at all levels of business and society before gender equality is achieved in the workplace', 'Young men need to learn from an early age that their male gender does not give them any superior rights compared to the female gender', 'Gender Equality']
8	1464	['ios', 'download', 'android', 'app', 'citizens', 'news', 'way', 'your', 'more', 'for']	['For more news your way download The Citizens app foriOS and Android', 'For more news your way download The Citizens app for iOS and Android', 'For more news your way download The Citizens app for iOS and Android']

9	1431	['she', 'her', 'shes', 'herself', 'career', 'actress', 'cyrus', 'mother', 'mary', 'daughter']	['Shes In Your Corner for consumer issues', 'We dont she said', 'It was her up against the world']
10	1155	['fire', 'fires', 'firefighters', 'wildfires', 'blaze', 'smoke', 'wildfire', 'burning', 'blazes', 'burned']	['I have never seen people trying to take out the fire with so many firefighters so much equipment and still be unable to contain the fire', 'With the fires you never know he said', 'They have the support of Working on Fire']
...	...	...	...
2892	15	['looting', 'kwazulunatal', 'riots', 'gauteng', 'looted', 'violent', 'unrest', 'quell', '15000', 'phoenix']	['Reflecting on the woes faced by KwaZuluNatal he cited the 2021 July looting and violent riots which had cost the province and the country billions of rand', 'Modise made reference to the July 2021 unrest and looting in Gauteng and KwaZuluNatal when an estimated 15000 members of the military were successfully deployed to quell the riots', 'Modise made reference to the July 2021 unrest and looting in Gauteng and KwaZuluNatal when an estimated 15000 members of the military were successfully deployed to quell the riots']

Table 2.3: Topic Distribution and Representative Documents

The table 2.3 presents the topics identified using the BERTopic framework, including the number of documents in each topic, the word representations generated through cTF-IDF and SBERT, and the main documents representing each topic. Topic '-1' denotes the outliers.

Some topic representations lack clarity; for example, words such as "he" and "she" in the outliers topic don't indicate whether the 207,940 documents contain meaningful data or just noise. To enhance this, we'll employ MMR for representation tuning in the next iteration. This approach will also tackle redundancies like "school" and "schools" or "recycled" and "recycling" found in topics 0 and 2. Additionally, understanding the outliers topic better should help us reduce its prominence.

Topic 8 reveals that 1,464 of our original 400,000 sentences are ads or mismatches, prompting us to add terms like "iOS" and "download" to our removal list.

Topic 1 is dominated by numbers, offering little insight. We've opted to exclude numbers during preprocessing since they provide minimal context. Although we could remove them during topic representation, doing so would leave noise like "00 7" and "00 11".

Although articles may cover multiple topics or be lengthy, they typically concentrate on a single topic per paragraph. As a result, we'll split the text by paragraph rather than by sentence or the entire document. Given that paragraphs can be extensive, we'll

employ "voyage-lite-02-instruct", a transformer optimized for classification, clustering, and sentence similarity tasks. Notably, "voyage-lite-02-instruct" achieves the state of the art performance in Semantic Textual Similarity according to the MTEB benchmark, and it can handle up to 4,000 tokens, whereas "all-MiniLM-L6-v2" has a 256-token limit.

With the shift to paragraph-level data splitting, we plan to incorporate data from various sources.

2.3.2 Second Model

Expanding upon the baseline model, we incorporated our newly preprocessed data, which excluded numbers and removed additional words indicative of spam or ads. Our dataset was structured at the paragraph level, encompassing a total of 1,425,684 paragraphs. For this iteration, we utilized the entire dataset with "voyage-lite-02-instruct" for embedding. Due to memory constraints, we opted for PCA over UMAP. We employed HDBSCAN for clustering, followed by cTF-IDF as our weighting scheme based on CountVectorizer. For representation tuning, we utilized MMR with a 0.3 similarity score.

Initially, we assessed the quality of PCA results by examining the explained variance: [0.00562548, 0.00540765, 0.00369897, 0.00341341, 0.00319785]. These values sum up to 0.021343354766047866, indicating that the five principal components accounted for only 2.13% of the total variance—this is notably low. Additionally, we evaluated the clustering quality using DBCV, yielding a score of 0.055616634424118, which is also not ideal.

Topic	Count	Representation	Representative_Docs
-1	1311370	['said', 'south', 'people', 'africa', 'government', 'time', 'police', 'year', 'years', 'african']	['Mzi Thebolla Msunduzi Municipality Mayor The people of South Africa, although belonging to many shades and culture, have learnt a lot from Nelson Mandela. One major lesson that we have learnt from Mandela is perseverance is that, in spite of years in jail, he persevered and never lost focus on what he wanted to achieve. This is a value the whole nation should cherish and know that whatever challenges may be encountered in the process of nation-building, perseverance will lead to success. . . . ']
0	11287	['citizen', 'original', 'story', 'read', '', '', '', '', '', '']	['Read original story on citizen.', 'Read original story on citizen.', 'Read original story on citizen.']
1	4573	['www', 'original', 'story', 'read', '', '', '', '', '', '']	['Read original story on www.', 'Read original story on www.', 'Read original story on www.']

2	2406	['informed', 'factually', 'factchecking', 'disinformation', 'filter', 'circulating', 'masses', 'adequately', 'credible', 'protocol']	['As your local news provider, we have the duty of keeping you factually informed on Covid- developments. As you may have noticed, mis- and disinformation (also known as fake news) is circulating online. Caxton Local Media is determined to filter through the masses of information doing the rounds and to separate truth from untruth in order to keep you adequately informed. Local newsrooms follow a strict pre-publication fact-checking protocol. A national task team has been established to assist in bringing you credible news reports on Covid-.', ...]
3	2406	['informed', 'factually', 'factchecking', 'disinformation', 'filter', 'circulating', 'masses', 'adequately', 'credible', 'protocol']	['As your local news provider, we have the duty of keeping you factually informed on Covid- developments. As you may have noticed, mis- and disinformation (also known as fake news) is circulating online. Caxton Local Media is determined to filter through the masses of information doing the rounds and to separate truth from untruth in order to keep you adequately informed. Local newsrooms follow a strict pre-publication fact-checking protocol. A national task team has been established to assist in bringing you credible news reports on Covid-.', ...]
4	2317	['queries', 'editor', 'adviser', 'readers', 'comments', 'contact', 'violet', 'legal', 'group', 'national']	['Readers with any comments or queries may contact National Group Editor Irma violet (irmacaxton.za) or Legal Adviser Helene Eloff (helenecaxton.', 'Readers with any comments or queries may contact National Group Editor Irma violet (irmacaxton.za) or Legal Adviser Helene Eloff (helenecaxton.', 'Readers with any comments or queries may contact National Group Editor Irma violet (irmacaxton.za) or Legal Adviser Helene Eloff (helenecaxton.']
5	2314	['queries', 'editor', 'adviser', 'readers', 'comments', 'contact', 'violet', 'legal', 'group', 'national']	['Readers with any comments or queries may contact National Group Editor Irma violet (irmacaxton.za) or Legal Adviser Helene Eloff (helenecaxton.', 'Readers with any comments or queries may contact National Group Editor Irma violet (irmacaxton.za) or Legal Adviser Helene Eloff (helenecaxton.', 'Readers with any comments or queries may contact National Group Editor Irma violet (irmacaxton.za) or Legal Adviser Helene Eloff (helenecaxton.']

6	2082	['combat', 'notice', 'aims', 'reporting', 'coronavirus', 'caxton', 'fake', 'media', 'news', 'local']	['Notice: Coronavirus reporting at Caxton Local Media aims to combat fake news', 'Notice: Coronavirus reporting at Caxton Local Media aims to combat fake news', 'Notice: Coronavirus reporting at Caxton Local Media aims to combat fake news']
7	2082	['combat', 'notice', 'aims', 'reporting', 'coronavirus', 'caxton', 'fake', 'media', 'news', 'local']	['Notice: Coronavirus reporting at Caxton Local Media aims to combat fake news', 'Notice: Coronavirus reporting at Caxton Local Media aims to combat fake news', 'Notice: Coronavirus reporting at Caxton Local Media aims to combat fake news']
8	1675	['engine', 'steering', 'litre', 'features', 'kmh', 'mm', 'interior', 'standard', 'models', 'seats']	['Globally the Fiesta is Fords bestselling car, and it does pretty well in SA too, where it competes most of the time with VWs Polo. And the eighth generation, all-new Fiesta was recently launched with a bag full of high tech, standard offering, and tricks to continue to battle its rivals. The all-new Ford Fiesta is said to raise the stakes in the compact car segment by building on the proud heritage of the very successful outgoing model, while lifting the benchmark significantly in terms of quality, refinement, technology and safety. The Fiestas widely acclaimed fun-to-drive character has been lifted to new heights, and its eye-catching new design embodies trend-setting style and sophistication. . . . ']
9	1309	['original', 'story', 'read', '', '', '', '', '', '', '']	['Read original story on rekordeast.', 'Read original story on rekordeast.', 'Read original story on rekordeast.']
10	1308	['original', 'story', 'read', '', '', '', '', '', '', '']	['Read original story on reviewonline.', 'Read original story on reviewonline.', 'Read original story on reviewonline.']
11	1269	['original', 'story', 'read', '', '', '', '', '', '', '']	['Read original story on lowvelder.', 'Read original story on lowvelder.', 'Read original story on lowvelder.']
12	1180	['rekord', 'original', 'story', 'read', '', '', '', '', '', '']	['Read original story on rekord.', 'Read original story on rekord.', 'Read original story on rekord.']
13	877	['websites', 'breaking', 'visit', 'free', 'community', 'news', 'flags', 'overseas', 'chinas', 'military']	['For free breaking and community news, visit Rekords websites:', 'For free breaking and community news, visit Rekords websites:', 'For free breaking and community news, visit Rekords websites:']
14	853	['original', 'story', 'read', '', '', '', '', '', '', '']	['Read original story on zululandobserver.', 'Read original story on zululandobserver.', 'Read original story on zululandobserver.']

15	737	['original', 'story', 'read', '', '', '', '', '', '', '']	['Read original story on northglennews.', 'Read original story on northglennews.', 'Read original story on northglennews.']
16	583	['original', 'story', 'read', '', '', '', '', '', '', '']	['Read original story on kemptonexpress.', 'Read original story on kemptonexpress.', 'Read original story on kemptonexpress.']
17	569	['original', 'story', 'read', '', '', '', '', '', '', '']	['Read original story on krugersdorpnews.', 'Read original story on krugersdorpnews.', 'Read original story on krugersdorpnews.']
18	546	['email', 'send', 'information', 'story', '', '', '', '', '', '']	['Do you have more information about the story? Please send us an email to editorialrekord.', 'Do you have more information about the story? Please send us an email to editorialrekord.', 'Do you have more information about the story? Please send us an email to editorialrekord.']
19	526	['original', 'story', 'read', '', '', '', '', '', '', '']	['Read original story on bedfordviewedenvalenews.', 'Read original story on bedfordviewedenvalenews.', 'Read original story on bedfordviewedenvalenews.']
...	...	...	...
1573	15	['risks', 'associated', 'estimates', 'mining', 'operations', 'related', 'politically', 'terrorism', 'compliance', 'imf']	['Needless to say, effects of the handshake included the UDA tsunami in the elections. Interestingly, while writings on the incidence of informality in the West have treated it largely positively, writings on informality in Africa, have defined it negatively, seeing as the cause of governance ailments. Informality in Africa is associated with corruption, where deals negotiated in the shadows lead to the capture of public procurement by politically connected elites and consequent loss of public funds. It is also associated with clientelism where the benefits of the state are parceled out to politically connected clients who also operate in the shadows.',...]

Table 2.4: Topic Distribution and Representative Documents for the Second Model

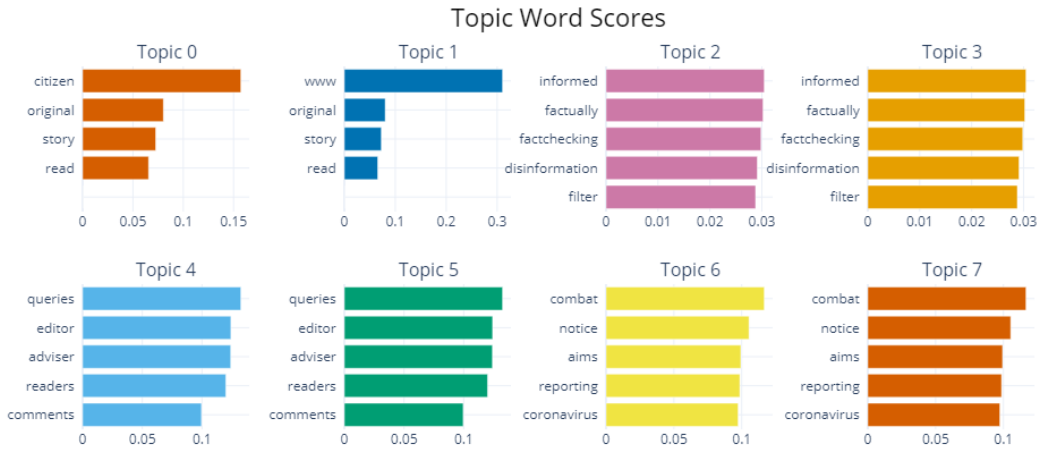


Fig. 2.5: Topic Word Scores for the top 8 topics of the second model

We examined the topic word scores through barplots (see Figure 2.5), revealing that the primary topics identified by this model were largely ads or spam. This observation was further confirmed by the similarity matrix (see Figure 2.6), which showed strong correlations among the identified topics. The topic distribution table 2.4 also indicated that the main topics predominantly originated from phrases like "Read original story on..." or similar end-of-article statements. Notably, the model categorized a significant portion of the data, 1,311,370 out of 1,425,684, as outliers, while the rest were not considered outliers only due to their frequent repetition. Consequently, we concluded that PCA was not an effective choice for this model. However, the model was not entirely without merit as it aided in identifying more spam and ads for removal, and in supporting the justification for choosing UMAP.

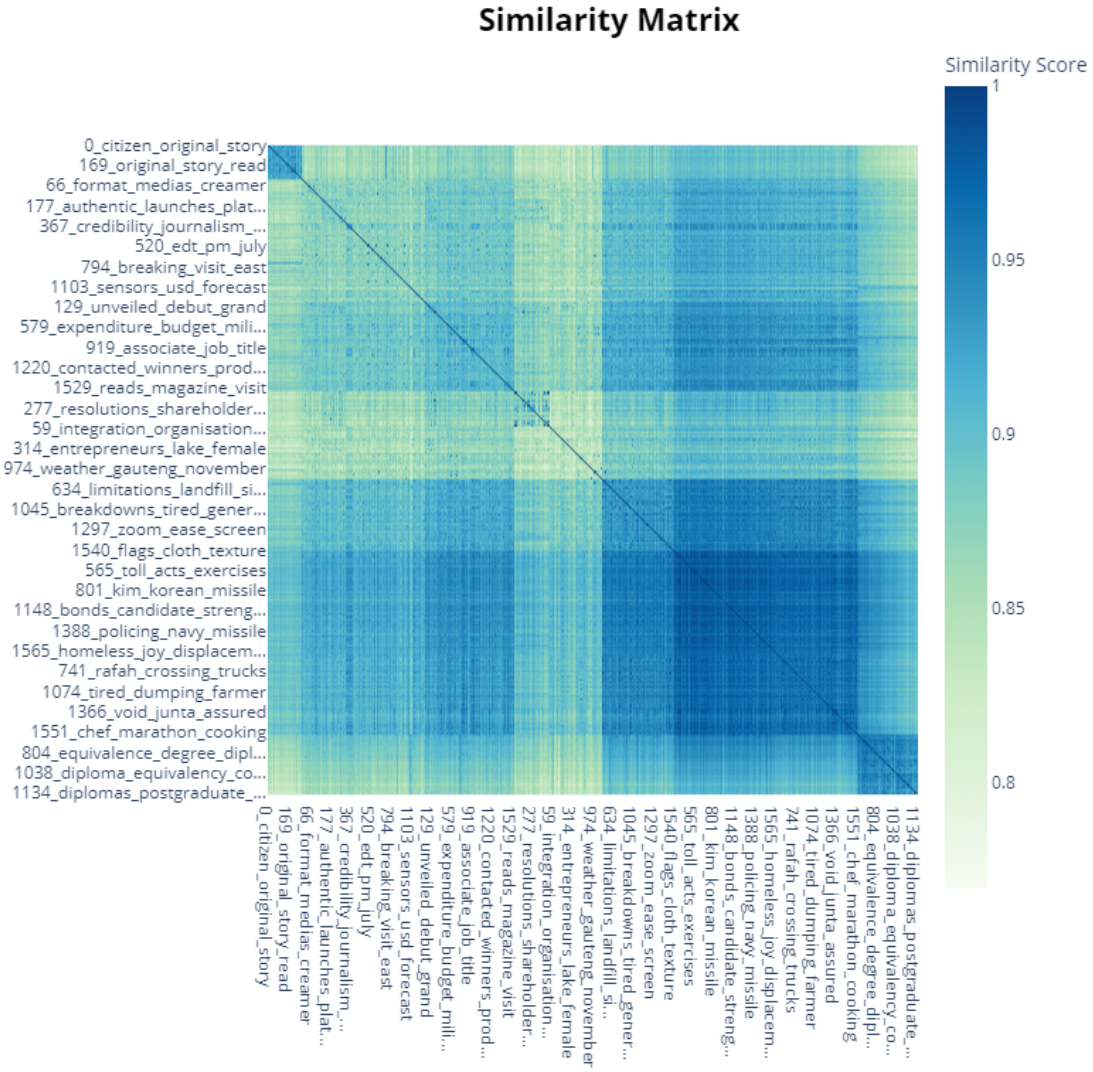


Fig. 2.6: Similarity Matrix for the topics of the second model

Given that we couldn't completely eliminate all spam/ads, our approach was to build a model that remains robust even in their presence, and then selectively remove the ones we identify. Additionally, due to hardware limitations preventing us from processing all 1.4M "voyage" embeddings, we planned to use UMAP on a subset of 500,000 paragraphs randomly selected from our dataset for the next iteration.

2.3.3 Third Model

For our third model, building on the second one and facing memory limitations, we sampled 500,000 paragraphs from the total of 1,425,684 paragraphs using a random state to ensure replicability and used UMAP. We continued to use "voyage-lite-02-instruct" for embeddings, HDBSCAN for clustering, and cTF-IDF as our weighting scheme, which is based on CountVectorizer. As for representation tuning, we maintained MMR with a

similarity score of 0.3.

Initially, we assessed the quality of the HDBSCAN clustering by examining the DBCV score, which was 0.22378513830608443. This score was notably better than that of the second model based on PCA, but there's room for improvement. Visually inspecting the similarity matrix of topics, we observed a distinct category of noise that was dissimilar to all other topics. Conversely, the remaining topics showed similarity, which is expected since our research is based on ESG and SDGs related keywords, which are often interconnected.

Topic	Count	Representation	Representative_Docs
-1	267334	['water', 'police', 'climate', 'said', 'people', 'court', 'change', 'public', 'government', 'told']	['In her Instagram dispatches, Palestinian journalist Bisan Owda often greets her . million followers with the same words: Hey everyone, this is Bisan from Gaza; Amid Israels ongoing bombardment of the Gaza Strip, her refrain serves as a reminder of the steep risks local reporters face to share unfiltered accounts of what is unraveling on the ground. at least Palestinian journalists have been killed since the war began, making it the deadliest war for reporters since recording commenced in , according to the Committee to Protect Journalists. While some of Owdas peers, including Motaz Azaiza, Plestia Alaqad, and Wael Dahdouh, have ultimately left Gaza for safety or medical care, the -year-old remains. . . . ', . . .]
0	7048	['schools', 'education', 'students', 'learners', 'teachers', 'pupils', 'school', 'universities', 'student', 'matric']	['According to the department, the pupils results will be available from their schools.', 'So while the opening of universities is not universal, we are certain that when we issue matric results on February next year, we will not be disadvantaging learners because most universities will start their academic year in March or April next year, she said.', . . .]
1	4490	['airline', 'airlines', 'aircraft', 'aviation', 'flights', 'flight', 'airport', 'airways', 'plane', 'saa']	['Best Airline Stock : United Airlines Holdings (UAL)', 'Acsa has even set up an aircraft noise committee - comprising members of the affected communities, representatives from the aviation industry, including aircraft control, airlines, civil aviation, and environmental groups - in a bid to tackle the problem.', 'There are more than of the Max alone in service around the world, and it can be found among the fleets of many major airlines, according to aviation analytics company Cirium. United Airlines has Max s in service, while Alaska Airlines has and plans for another of the aircraft.']

2	4438	['zuma', 'anc', 'ancs', 'jacob', 'dlaminizuma', 'zumas', 'nec', 'conference', 'party', 'partys']	['ANCs president Jacob Zuma, and deputy president Cyril Ramaphosa lead the NEC in song during the partys Siyanqoba election rally, FNB Stadium May .', 'President Cyril Ramaphosa has warned members of the ANC national executive committee (NEC) to be cautious when addressing issues about candidates on the partys election list because voters were angry.Sources who attended the ANCs special NEC meeting last Monday at Saint Georges Hotel in Tshwane said Ramaphosa cut short what would have been an overnight discussion about the controversial list.', ...]
3	3989	['citizen', 'original', 'story', 'read', 'send', 'submit', 'app', 'email', 'reporting', 'digital']	['Read original story on citizen.', 'Read original story on citizen.', 'Read original story on citizen.']
4	2985	['luck', 'im', 'bad', 'things', 'ive', 'thoughts', 'happen', 'think', 'thank', 'humility']	['Some people have all the luck. Have you ever had this thought? Its time to get rid of it once and for all. Those people arent blessed by luck they have made conscious decisions to do these nine things.', 'Successful people know that the answer to Why do I have such bad luck? doesnt matter, because they control their own destinies. They have a growth mindset and believe that change is possible. They take responsibility for their lives and believe that they alone can make it a masterpiece. This is called having an internal locus of control, and its essential to turning your luck around.', ...]
5	2273	['hospitals', 'healthcare', 'nurses', 'hospital', 'doctors', 'health', 'medical', 'patients', 'beds', 'schemes']	['The department was allocated R. billion to employ healthcare workers and posts, including professional nurses, staff nurses, administration clerks, and general orderlies, among others, were filled, the MEC said.', 'Ngwenya said employing doctors and nurses was not the only solution because currently many public hospitals did not have the facilities or equipment to provide quality healthcare to patients.', ...]

6	2169	['gender', 'women', 'womens', 'equality', 'empowerment', 'female', 'representation', 'equal', 'inclusion', 'parity']	['Challenges and opportunities in achieving gender equality and the empowerment of rural women and girls;', 'Joint Programme on Gender Equality and Women Empowerment', 'Xingwana said a process of developing womens empowerment and gender equality policy that would lead to the Women Empowerment and Gender Equality Bill was at an advanced stage.']
...	...	...	...
2474	15	['peacekeepers', 'peace', 'aided', 'peacekeeping', 'commended', 'subregion', 'preserving', 'troops', 'restore', 'spelt']	['Aroldo Lzaro, the Head of Mission and Force Commander, commended the work of the more than , military peacekeepers from countries and the civilian staff.', 'According to him, All Nigerian peacekeepers alongside others from around the world have aided the course of humanity while helping nations in distress to restore peace and enthrone much desired development.', 'All Nigerian peacekeepers alongside others from around the world have aided the course of humanity, while helping nations in distress to restore peace and enthrone much-desired development.']

Table 2.5: Topic Distribution and Representative Documents for the Third Model

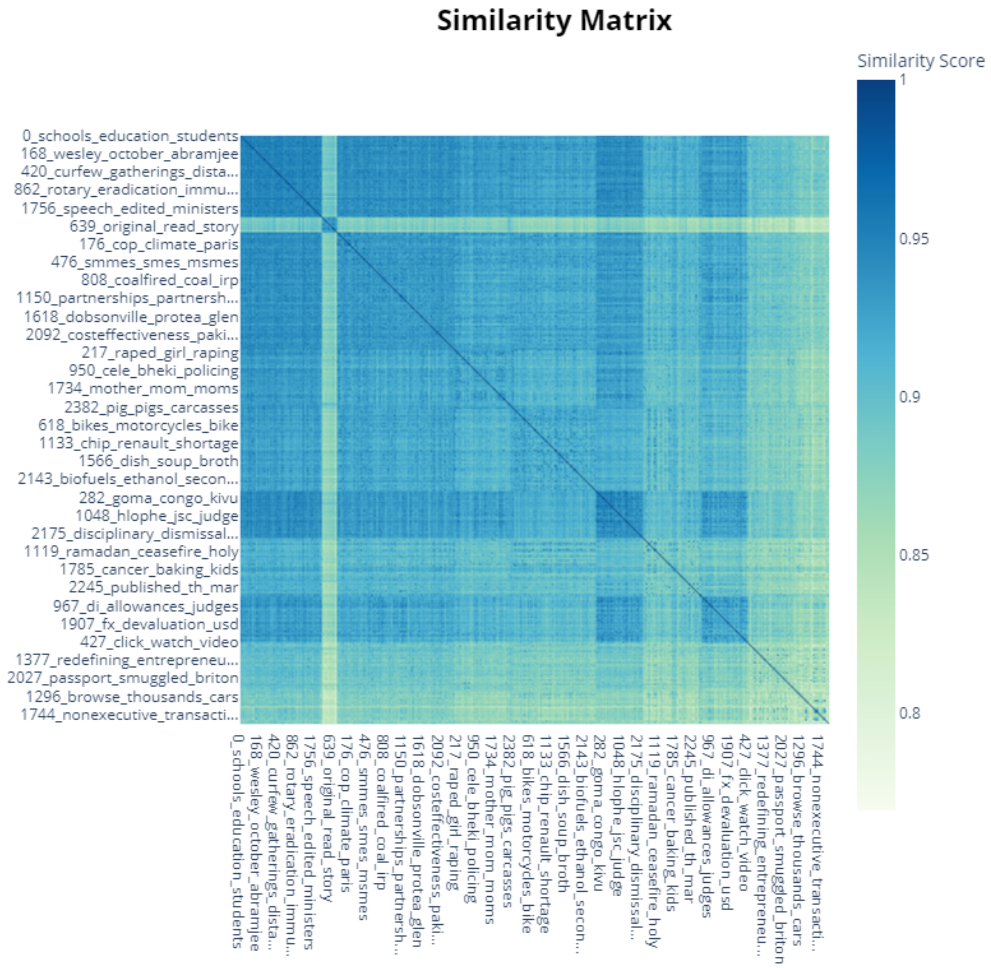


Fig. 2.7: Similarity Matrix for the topics of the Third model

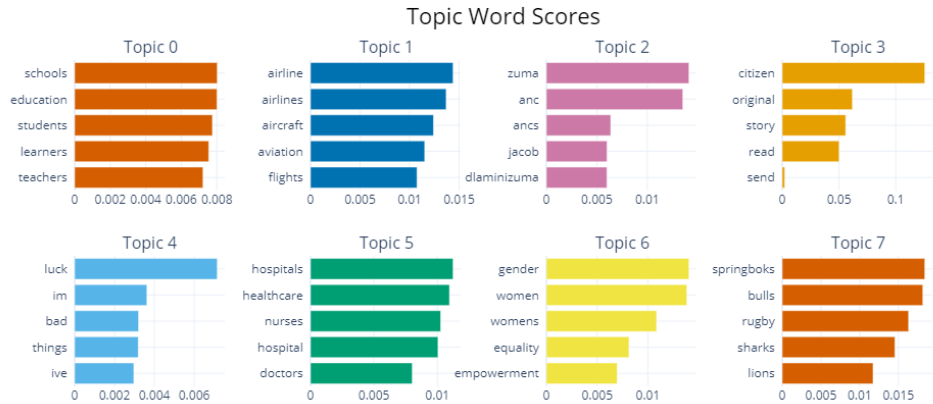


Fig. 2.8: Topic Word Scores for the top 8 topics of the third model

Topics clearly related to spam, like Topic 3, can be disregarded. However, some topics remain challenging to interpret, such as Topic 4, represented by words like "luck," "im," "bad," "things," and "ive," which offer little insight into its meaning. Additionally, we noticed redundancies in topic representations, like "airline" and "airlines." To address these issues, we plan to increase the similarity score in representation tuning. Nonetheless, we prioritize addressing these main challenges first, as representation tuning can be fine-tuned later once we are satisfied with our model. Its impact on the quality of sentiment analysis is relatively minor.

For our next iteration, we plan to implement binary quantization to reduce data memory usage. We'll use the same 500,000 paragraphs to assess the impact on quality using our current parameters. This step prepares us for applying the technique to the entire dataset. We aim to understand how binary quantization affects our model, specifically whether applying it to embeddings significantly reduces model quality. If the impact is minimal, we'll proceed to use it on the complete dataset to train our model.

2.3.4 Fourth Model

For the fourth model, we'll maintain the same methods as in the last model with the addition of binary quantization. We'll start with the same 500,000 randomly selected paragraphs, extract their voyage embeddings, and apply binary quantization for compression. We'll then use UMAP for dimensionality reduction, HDBSCAN for clustering, and c-TFIDF based on CountVectorizer for topic representation. Finally, we'll use MMR with a similarity score of 0.3. All hyperparameters will remain the same as those used in the third model.

We started by assessing the quality of the HDBSCAN clustering by examining the DBCV score, which was 0.190765434699169 which isn't such a loss compared to the model with the non quantized embeddings which was 0.22378513830608443.

Topic	Count	Representation
-1	291078	['said', 'police', 'government', 'people', 'public', 'state', 'court', 'business', 'president', 'economic']
0	17718	['cup', 'players', 'rugby', 'league', 'coach', 'games', 'match', 'football', 'game', 'tournament']
1	11045	['engine', 'rear', 'car', 'electric', 'cars', 'vehicles', 'models', 'vehicle', 'ford', 'wheels']
2	4001	['airline', 'airlines', 'aircraft', 'aviation', 'flights', 'flight', 'airport', 'plane', 'airways', 'saa']
3	3996	['zuma', 'anc', 'party', 'jacob', 'ancs', 'dlaminizuma', 'zumas', 'partys', 'ramaphosa', 'conference']
4	3976	['citizen', 'original', 'story', 'read', 'onthe', 'drafted', 'conflicts', 'advised', 'circumstances', 'sent']
5	3684	['pepper', 'sauce', 'add', 'chicken', 'salt', 'chopped', 'butter', 'cook', 'cheese', 'dish']
6	3558	['music', 'album', 'song', 'songs', 'musical', 'band', 'jazz', 'musicians', 'artists', 'artist']
7	3380	['gaza', 'israel', 'israeli', 'hamas', 'ceasefire', 'palestinian', 'palestinians', 'israels', 'aid', 'hostages']
8	3232	['inflation', 'ksh', 'shares', 'banks', 'rate', 'bank', 'investors', 'net', 'equity', 'dollar']
9	2734	['racing', 'race', 'class', 'championship', 'races', 'bike', 'formula', 'rally', 'max', 'riders']
10	2675	['comedy', 'film', 'drama', 'starring', 'thriller', 'movie', 'love', 'novel', 'characters', 'series']
11	2581	['farmers', 'agricultural', 'agriculture', 'farming', 'maize', 'food', 'crops', 'crop', 'smallholder', 'irrigation']
12	2333	['women', 'gender', 'womens', 'equality', 'empowerment', 'female', 'inclusion', 'gap', 'equal', 'girls']
13	2149	['rhino', 'poaching', 'elephants', 'lion', 'elephant', 'lions', 'horn', 'leopard', 'wildlife', 'animals']
14	1775	['waste', 'plastic', 'recycling', 'plastics', 'landfill', 'recycled', 'bags', 'recycle', 'packaging', 'dumping']
15	1773	['property', 'buyers', 'rental', 'tenants', 'properties', 'estate', 'landlord', 'sellers', 'tenant', 'rent']
16	1617	['www', 'original', 'story', 'read', 'breaks', 'bbc', 'silence', 'investor', 'japan', 'gain']
17	1570	['child', 'kids', 'parents', 'children', 'childs', 'parenting', 'fun', 'parent', 'play', 'toddler']
18	1489	['cyber', 'cybersecurity', 'security', 'password', 'passwords', 'data', 'attacks', 'fraud', 'computer', 'software']
19	1388	['flowers', 'plants', 'leaves', 'plant', 'soil', 'flower', 'garden', 'watering', 'flowering', 'grow']
...	...	...
1329	15	['pm', 'opening', 'holidays', 'anchor', 'open', 'reservations', 'likewise', 'hours', 'feb', 'fridays']

Table 2.6: List of topics with their count and word representation for the fourth model

By referring to the topic list in Table 2.6, we can conduct a comparative analysis with the third model to examine the impact of quantization on our topic clustering. Initially, we observe an expansion in the size of the outliers topic, growing from 267,334 paragraphs to 291,078. While this might initially suggest a degradation in model quality, it's important to note that the quantization-induced loss could potentially act as a regularization step, unexpectedly bolstering the robustness of the topic modeling process.

Furthermore, we note a notable change in topic distribution between the third and fourth models. Specifically, topic 0 in the third model, primarily addressing education with a total of 7,048 paragraphs, appears to have been lost in our fourth model, despite employing the same dataset. Additionally, the topic related to airlines and flights (topic 1) from the third model has now emerged as the third topic in the fourth model, indicating a shift in the topics being modeled.

Moreover, the reduction in the total number of topics from 2474 to 1329 suggests potential fusion of some topics due to less precise embeddings. This consolidation could indicate a refinement or simplification in the topic structure, potentially enhancing interpretability or reducing redundancy within the model.

Given the impracticality of examining each individual topic, our assessment of the decline in topic modeling quality from the third model to the fourth model relies on the objective evaluation provided by the DBCV metric. Consequently, we can assert that while there may have been a slight increase in outliers, the overall reduction in the quality of topic modeling was minimal.

2.3.5 Fifth Model

For our fifth model, we'll implement binary quantization based on its promising results from the previous model, which showed minimal quality degradation in the DBCV score. Reflecting on our past models, we've made the following decisions:

- Given that paragraphs better capture topics than sentences or entire texts, we'll utilize Voyage for its capability to handle 4,000 tokens and its state-of-the-art performance in semantic similarity tasks.
- PCA proved to be ill-suited for our needs based on the second model. In contrast, both research papers and our third model indicate UMAP as the superior choice for dimensionality reduction.
- For clustering, HDBSCAN will be our method of choice, as k-means struggles with non-spherical data shapes.
- We'll continue using CountVectorizer and c-TFIDF for topic representation and MMR to improve word representations.
- We'll experiment with different UMAP, HDBSCAN parameters using the DBCV score as an evaluation metric.

We will utilize a randomly selected subset of 200,000 paragraphs from our dataset to optimize hyperparameters for our BERTOPIC framework. Our primary focus will be on UMAP and HDBSCAN. While topic representation serves as a valuable tool for visually evaluating the model, it remains independent of the dimensionality reduction and clustering phases.

Although we could assess the trustworthiness to validate the dimensionality reduction process before evaluating HDBSCAN using DBCV, our primary interest lies in the clustering results. Additionally, retaining 10 dimensions instead of 5 with UMAP is likely to yield better results, while HDBSCAN may be adversely affected by higher dimensions. Therefore, the combined effects of these processes are more indicative of result quality.

Furthermore, assessing trustworthiness across different UMAP models poses challenges due to the inability to use varying max_k values for different models, as they select different numbers of neighbors. With approximately 96 models resulting from our hyperparameter grid, each requiring about an hour to produce results, the process becomes highly time-consuming, with limited potential for parallelization due to its computational intensity.

Regarding the quantization step, literature suggests the use of the Hamming distance as an efficient metric equivalent to cosine similarity. To reduce memory usage, we pack 0s and 1s into bits, consolidating every 8 elements into a single unsigned integer (uint8) number ranging from 0 to 255. However, a challenge arises as the Hamming distance yields a distance of 1 whenever two numbers differ. For instance, in a 4-dimensional space, consider the vectors (1, 1, 1, 1), (1, 1, 1, 0), and (0, 0, 0, 0), transformed into (15), (14), and (0) respectively. Directly applying the Hamming distance to [(15), (14)] and [(15), (0)] would yield identical results of 1. However, when applied to the original data, the distances would be 1 and 4 respectively. This discrepancy underscores another lossy aspect introduced by the quantization process. To mitigate this issue, we will adjust the Hamming distance to directly apply to uint8 vectors, thereby ensuring consistent results with the original quantized embeddings.

We have optimized our selection of hyperparameters from UMAP and HDBSCAN to achieve the highest DBCV scores. Specifically, for UMAP, we have carefully selected values for 'n_neighbors' from 5, 10, 20, 50, 100, 200, and 500, 'n_components' from 5 to 10, 'min_dist' from 0.0, 0.1, and 0.25, and the metric, which can either be the Hamming distance or a custom adjusted Hamming distance named 'Ghali'. Regarding HDBSCAN, we have explored 'min_cluster_size' options of 10, 15, 30, 60, and 150, and 'min_samples' options of 10, 15, 30, 60, and 150.

n neighbors	n components	min dist	metric	min cluster size	min samples	DBCV
500	5	0.0	hamming	150	30	0.885642
5	5	0.25	hamming	60	150	0.765343
5	5	0.25	hamming	10	150	0.761654
5	5	0.25	hamming	30	150	0.761594
5	5	0.25	hamming	15	150	0.761594
5	5	0.25	hamming	150	150	0.761269
10	10	0.0	hamming	10	150	0.294039
10	10	0.0	hamming	15	150	0.292358
10	10	0.0	hamming	30	150	0.290661
10	10	0.0	hamming	60	150	0.290276
20	10	0.0	ghali	30	150	0.285165
10	10	0.0	hamming	150	150	0.281002
50	5	0.1	hamming	150	15	0.276733
20	10	0.0	ghali	150	150	0.270798
20	10	0.0	hamming	150	10	0.269222

Table 2.7: DBCV score for different hyperparameters. Similar models have the same color.

We've explored 2400 potential combinations of hyperparameters for our model. The optimal configuration delivers an impressive DBCV score of 0.885642, achieved with the following settings: `n_neighbors=500`, `n_components=5`, `min_dist=0.0`, `metric=Hamming`, `min_cluster_size=30`, and `min_samples=30`. However, we've observed significant discrepancies when varying the min_samples parameter, resulting in scores of 0.037587, 0.016503, 0.044853, and 0.036217 respectively (see Table 2.8). This variance undermines our confidence in the effectiveness of the hyperparameters.

n neighbors	n components	min dist	metric	min cluster size	min samples	DBCV
500	5	0.0	hamming	150	10	0.037587
500	5	0.0	hamming	150	15	0.016503
500	5	0.0	hamming	150	30	0.885642
500	5	0.0	hamming	150	60	0.044853
500	5	0.0	hamming	150	150	0.036217

Table 2.8: Variation in DBCV Scores Based on Min Samples Variance of the 0.88 Model

We examine the topics derived from the model with a DBCV score of 0.885642 to identify any potential issues. We discovered that while the outliers topic is nearly vacant, the remaining topics are predominantly nonsensical (refer to Table 2.9), likely originating from spam. Consequently, this renders the modeling as ineffective as others, with a DBCV lower than 0.05.

Topic	Count	Representation
-1	1407	['original', 'story', 'read', '', '', '', '', '', '', '']
0	191963	['read', 'story', 'original', '', '', '', '', '', '', '']
1	1623	['original', 'story', 'read', '', '', '', '', '', '', '']
2	664	['', '', '', '', '', '', '', '', '', '']
3	657	['story', 'read', '', '', '', '', '', '', '', '']
4	631	['original', 'story', 'read', '', '', '', '', '', '', '']
5	518	['original', 'story', 'read', '', '', '', '', '', '', '']
6	427	['original', 'story', 'read', '', '', '', '', '', '', '']
7	388	['', '', '', '', '', '', '', '', '', '']
8	315	['original', 'story', 'read', '', '', '', '', '', '', '']
9	278	['', '', '', '', '', '', '', '', '', '']
10	255	['', '', '', '', '', '', '', '', '', '']
11	252	['original', 'story', 'read', '', '', '', '', '', '', '']
12	243	['original', 'story', 'read', '', '', '', '', '', '', '']
13	207	['original', 'story', 'read', '', '', '', '', '', '', '']
14	172	['original', 'story', 'read', '', '', '', '', '', '', '']

Table 2.9: Topic Representation for the model with DBCV Score of 0.885642

For the following hyperparameters from Table 2.7, we noticed that they shared the same `n_neighbors=5`, `n_components=5`, `min_dist=0.25`, and `metric=Hamming` for the UMAP part and the same `min_samples=150`, along with nearly the same DBCV score of approximately 0.76. So we checked manually only one, the same way as before using the topic representation, since they didn't differ by much.

The outcomes were equally disappointing compared to those of the model with the 0.88 DBCV score (see Table 2.10), indicating that these hyperparameters are unsuitable. It became evident that the topics captured by the model were predominantly spam. Furthermore, the cTF-IDF representation failed to find relevant keywords for the last topic.

Continuing with our exploration of hyperparameters, we find that for the UMAP phase, we have set `n_neighbors=10`, `n_components=10`, `min_dist=0.0`, and `metric=Hamming`. Similarly, for the HDBSCAN clustering step, we maintain `min_samples=150`, aiming for a smaller `min_cluster_size` for optimal performance, albeit with only marginal improvements. Referring to Table 2.11, we observe that while more than half of our documents are allocated to the outliers topic, the remaining topics exhibit promising word representations. Some topics may benefit from further representation tuning, such as topic 2, which appears to center around animals. However, since our focus is primarily on evaluating the dimensionality reduction and clustering aspects of our framework, we defer addressing this issue for the time being. Thus, we will test these hyperparameters in our final model.

Topic	Count	Representation
-1	4421	['original', 'story', 'read', 'news', 'local', 'media', '', '', '', '']
0	188248	['media', 'local', 'news', 'read', 'story', 'original', '', '', '', '']
1	1692	['news', 'media', 'local', 'read', 'story', '', '', '', '', '']
2	1620	['original', 'story', 'read', '', '', '', '', '', '', '']
3	973	['original', 'story', 'read', 'news', '', '', '', '', '', '']
4	603	['original', 'story', 'read', 'news', 'local', 'media', '', '', '', '']
5	349	['media', 'news', 'local', 'story', 'original', 'read', '', '', '', '']
6	342	['media', 'news', 'local', '', '', '', '', '', '', '']
7	330	['local', 'media', 'news', 'read', '', '', '', '', '', '']
8	312	['news', 'local', 'media', 'original', 'read', 'story', '', '', '', '']
9	302	['news', 'local', 'media', '', '', '', '', '', '', '']
10	291	['media', 'news', 'local', '', '', '', '', '', '', '']
11	184	['original', 'story', 'read', '', '', '', '', '', '', '']
12	177	['original', 'story', 'read', 'news', '', '', '', '', '', '']
13	92	['original', 'story', 'read', '', '', '', '', '', '', '']
14	64	['', '', '', '', '', '', '', '', '', '']

Table 2.10: Topic Representation for the model with DBCV Score of 0.765343

Topic	Count	Representation
-1	113385	['said', 'people', 'south', 'government', 'court', 'police', 'africa', 'public', 'new', 'years']
0	6984	['cup', 'players', 'rugby', 'game', 'league', 'match', 'team', 'games', 'football', 'coach']
1	6642	['car', 'vehicle', 'engine', 'cars', 'vehicles', 'rear', 'electric', 'model', 'new', 'models']
2	4906	['animals', 'species', 'rhino', 'wildlife', 'animal', 'dogs', 'conservation', 'wild', 'dog', 'park']
3	4104	['health', 'vaccine', 'tb', 'disease', 'patients', 'covid', 'healthcare', 'hospital', 'virus', 'medical']
4	3553	['suspects', 'police', 'vehicle', 'shot', 'men', 'armed', 'suspect', 'arrested', 'stolen', 'scene']
5	3122	['climate', 'food', 'change', 'farmers', 'agriculture', 'agricultural', 'global', 'drought', 'temperatures', 'warming']
6	3037	['music', 'album', 'song', 'songs', 'love', 'film', 'life', 'artists', 'comedy', 'like']
7	2961	['bank', 'debt', 'investors', 'banks', 'capital', 'shares', 'investment', 'financial', 'market', 'ksh']
8	2905	['anc', 'zuma', 'party', 'president', 'jacob', 'ramaphosa', 'conference', 'ancs', 'leader', 'da']
9	2606	['education', 'school', 'students', 'schools', 'teachers', 'learners', 'language', 'university', 'pupils', 'learning']
10	2252	['women', 'gender', 'womens', 'violence', 'equality', 'genderbased', 'girls', 'men', 'female', 'empowerment']
11	1688	['airline', 'airport', 'aircraft', 'aviation', 'flight', 'air', 'flights', 'plane', 'airways', 'saa']
12	1657	['add', 'pepper', 'salt', 'chicken', 'minutes', 'butter', 'heat', 'cook', 'cheese', 'pan']
13	1622	['citizen', 'original', 'story', 'read', '', '', '', '', '', '']
14	1522	['rape', 'raped', 'mother', 'child', 'girl', 'allegedly', 'man', 'yearold', 'police', 'victim']
15	1352	['sudan', 'conflict', 'forces', 'peace', 'sudanese', 'ethiopia', 'humanitarian', 'war', 'military', 'rsf']
16	1303	['gaza', 'israel', 'israeli', 'aid', 'palestinian', 'palestinians', 'israels', 'war', 'humanitarian', 'resolution']
17	1275	['water', 'sanitation', 'river', 'supply', 'dam', 'residents', 'drinking', 'pollution', 'city', 'municipality']
18	1259	['child', 'children', 'parents', 'kids', 'childs', 'play', 'time', 'skills', 'parent', 'behaviour']
19	1250	['scene', 'injuries', 'injured', 'accident', 'hospital', 'driver', 'truck', 'ambulance', 'vehicle', 'transported']
...	...	...
116	12	['gupta', 'guptas', 'stateowned', 'family', 'zuma', 'familys', 'zondo', 'boss', 'involvement', 'entities']

Table 2.11: Topic Representation for the model with DBCV Score of 0.294039

Next, we'll examine the top-performing models utilizing the adjusted Hamming distance (Ghali). We observe that these models share specific hyperparameters: `n_neighbors=20`, `n_components=10`, `min_dist=0.0`, and `min_samples=150`, with slight fluctuations in the DBCV score corresponding to adjustments in the `min_cluster_size` parameter (refer to Table 2.12).

n neighbors	n components	min dist	metric	min cluster size	min samples	DBCV
20	10	0.0	ghali	30	150	0.285165
20	10	0.0	ghali	150	150	0.270798
20	10	0.0	ghali	15	150	0.268265
20	10	0.0	ghali	10	150	0.262944
20	10	0.0	ghali	60	150	0.186481

Table 2.12: Variation in DBCV Scores Based on Min Cluster Size Variance of the Adjusted Hamming Distance Models

Referring to Table 2.13, we observe that while more than half of our documents are allocated to the outliers topic, the remaining topics exhibit promising word representations. Thus, we will test these hyperparameters in our final model.

Topic	Count	Representation
-1	112720	['said', 'people', 'south', 'government', 'africa', 'new', 'public', 'years', 'year', 'time']
0	20856	['police', 'women', 'suspects', 'arrested', 'said', 'man', 'scene', 'men', 'vehicle', 'murder']
1	5440	['cup', 'players', 'league', 'game', 'rugby', 'match', 'coach', 'team', 'football', 'win']
2	4974	['animals', 'species', 'wildlife', 'rhino', 'animal', 'conservation', 'dogs', 'dog', 'wild', 'park']
3	4276	['car', 'engine', 'vehicle', 'rear', 'vehicles', 'cars', 'electric', 'model', 'models', 'new']
4	3630	['education', 'students', 'school', 'schools', 'university', 'teachers', 'learners', 'degree', 'grade', 'pupils']
5	3395	['energy', 'eskom', 'power', 'electricity', 'gas', 'renewable', 'solar', 'oil', 'nuclear', 'generation']
6	3237	['bank', 'banks', 'debt', 'investors', 'investment', 'shares', 'capital', 'financial', 'market', 'billion']
7	2682	['music', 'song', 'film', 'love', 'band', 'musical', 'best', 'artists', 'life', 'like']
8	2556	['climate', 'change', 'farmers', 'food', 'agriculture', 'agricultural', 'global', 'temperatures', 'land', 'warming']
9	1771	['anc', 'zuma', 'party', 'president', 'ramaphosa', 'conference', 'da', 'partys', 'jacob', 'provincial']
10	1740	['airline', 'aircraft', 'airport', 'flight', 'aviation', 'flights', 'air', 'plane', 'airways', 'passengers']
11	1622	['citizen', 'original', 'story', 'read', ", ", ", ", ", ", "]
12	1509	['add', 'pepper', 'salt', 'minutes', 'chicken', 'heat', 'butter', 'cook', 'pan', 'taste']
13	1468	['water', 'dam', 'sanitation', 'supply', 'river', 'drinking', 'residents', 'drought', 'rivers', 'access']
14	1306	['gaza', 'israel', 'israeli', 'aid', 'palestinian', 'humanitarian', 'war', 'resolution', 'military', 'strip']
15	1132	['sudan', 'military', 'forces', 'sudanese', 'ethiopia', 'niger', 'conflict', 'peace', 'region', 'war']
16	1110	['vaccine', 'tb', 'vaccines', 'vaccination', 'disease', 'health', 'covid', 'malaria', 'ebola', 'virus']
17	1041	['child', 'children', 'parents', 'kids', 'childs', 'play', 'time', 'help', 'skills', 'behaviour']
18	905	['games', 'olympic', 'sports', 'sport', 'gold', 'championships', 'world', 'marathon', 'race', 'sporting']
19	768	['cyber', 'data', 'security', 'cybersecurity', 'fraud', 'information', 'attacks', 'breach', 'email', 'identity']
...	...	...
90	36	['tourism', 'tourists', 'visa', 'domestic', 'sa', 'industry', 'international', 'travel', 'tourist', 'sector']

Table 2.13: Topic Representation for the best model in terms of DBCV score, utilizing the adjusted Hamming metric.

The next set of hyperparameters we're examining consists of `n_neighbors=50`, `n_components=5`, `min_dist=0.1`, `metric=hamming`, `min_cluster_size=150`, and `min_samples=15`. Initially, we seek similar hyperparameter configurations with commendable DBCV scores, aiming to discern whether this instance mirrors the earlier case of overfitting with the 0.88 model or represents a distinct category of hyperparameters. However, the results present a conundrum: while `min_samples=15, 10, 30` yield relatively favorable outcomes, `min_samples=60, 150` perform inadequately (refer to Table 2.14). Hence, we need to revisit the topic distributions to determine the viability of testing these hyperparameters in our final model. Referring to Table 2.15, we note another disappointing model characterized by inadequate word representations, with some topics barely comprising ten words, let alone offering a comprehensive representation. Consequently, we will omit testing these hyperparameters in our final model.

n neighbors	n components	min dist	metric	min cluster size	min samples	DBCV
50	5	0.1	hamming	150	15	0.276733
50	5	0.1	hamming	150	10	0.256693
50	5	0.1	hamming	150	30	0.230415
50	5	0.1	hamming	150	60	0.036223
50	5	0.1	hamming	150	150	0.026069

Table 2.14: Variation in DBCV Scores Based on Min Samples Variance of the 0.276733 Model.

Topic	Count	Representation
-1	909	['original', 'story', 'read', 'news', '', '', '', '', '', '']
0	192485	['news', 'read', 'story', 'original', '', '', '', '', '', '']
1	1620	['original', 'story', 'read', '', '', '', '', '', '', '']
2	629	['original', 'story', 'read', '', '', '', '', '', '', '']
3	532	['story', 'read', 'news', 'original', '', '', '', '', '', '']
4	378	['original', 'story', 'read', 'news', '', '', '', '', '', '']
5	362	['original', 'story', 'read', 'news', '', '', '', '', '', '']
6	360	['news', '', '', '', '', '', '', '', '', '']
7	320	['news', 'story', '', '', '', '', '', '', '', '']
8	319	['news', 'story', '', '', '', '', '', '', '', '']
9	319	['news', 'read', '', '', '', '', '', '', '', '']
10	303	['original', 'story', 'read', 'news', '', '', '', '', '', '']
11	257	['original', 'story', 'read', '', '', '', '', '', '', '']
12	247	['original', 'story', 'read', '', '', '', '', '', '', '']
13	217	['original', 'story', 'read', '', '', '', '', '', '', '']
14	208	['original', 'story', 'read', '', '', '', '', '', '', '']
15	185	['original', 'story', 'read', '', '', '', '', '', '', '']
16	182	['original', 'story', 'read', '', '', '', '', '', '', '']
17	168	['original', 'story', 'read', '', '', '', '', '', '', '']

Table 2.15: Topic Representation for the model with DBCV Score of 0.276733

The final set of hyperparameters we are testing comprises `n_neighbors=20`, `n_components=10`, `min_dist=0.0`, `metric=hamming`, `min_cluster_size=150`, and `min_samples=10`. Upon

reviewing the topics through a preliminary manual assessment, we observed potential issues with spam-related representations in topics 11 and 105. This discrepancy may stem from the min_cluster_size parameter being too large. Nevertheless, on a broader scale, the representations largely conform to our expectations (see Table 2.16). Consequently, we plan to evaluate this hyperparameter configuration in our final model.

Topic	Count	Representation
-1	119979	['said', 'people', 'south', 'government', 'africa', 'public', 'years', 'new', 'time', 'court']
0	7473	['cup', 'players', 'game', 'league', 'team', 'match', 'games', 'coach', 'football', 'win']
1	6001	['car', 'vehicle', 'engine', 'cars', 'vehicles', 'electric', 'new', 'model', 'models', 'toyota']
2	4426	['animals', 'species', 'rhino', 'wildlife', 'animal', 'dogs', 'conservation', 'dog', 'wild', 'park']
3	4184	['energy', 'eskom', 'power', 'gas', 'oil', 'electricity', 'renewable', 'solar', 'loadshedding', 'fuel']
4	3602	['suspects', 'police', 'vehicle', 'shot', 'arrested', 'suspect', 'men', 'armed', 'stolen', 'scene']
5	3158	['road', 'taxi', 'transport', 'traffic', 'drivers', 'motorists', 'driver', 'scene', 'roads', 'accident']
6	3052	['anc', 'zuma', 'party', 'president', 'jacob', 'ramaphosa', 'conference', 'da', 'leader', 'deputy']
7	2382	['police', 'body', 'man', 'mother', 'yearold', 'rape', 'child', 'allegedly', 'victim', 'girl']
8	2087	['dividend', 'bank', 'debt', 'inflation', 'investment', 'financial', 'ksh', 'investors', 'banks', 'capital']
9	1871	['music', 'songs', 'song', 'band', 'artists', 'film', 'love', 'stage', 'best', 'live']
10	1697	['airlines', 'aircraft', 'airport', 'flight', 'aviation', 'flights', 'air', 'plane', 'airways', 'passengers']
11	1623	['citizen', 'original', 'story', 'read', 'verify', 'working', '', '', '', '']
12	1593	['sudan', 'niger', 'military', 'forces', 'sudanese', 'ethiopia', 'conflict', 'burkina', 'peace', 'region']
13	1575	['add', 'pepper', 'salt', 'chicken', 'minutes', 'heat', 'cook', 'butter', 'pan', 'chocolate']
14	1515	['vaccine', 'tb', 'disease', 'vaccines', 'vaccination', 'health', 'blood', 'covid', 'malaria', 'virus']
15	1447	['education', 'school', 'students', 'schools', 'teachers', 'learners', 'pupils', 'department', 'grade', 'university']
16	1370	['gaza', 'israel', 'israeli', 'aid', 'palestinian', 'palestinians', 'israels', 'war', 'resolution', 'humanitarian']
17	945	['women', 'gender', 'womens', 'equality', 'empowerment', 'female', 'men', 'leadership', 'girls', 'economic']
18	922	['work', 'unemployment', 'job', 'youth', 'skills', 'employment', 'jobs', 'employees', 'employers', 'young']
19	856	['child', 'children', 'parents', 'kids', 'childs', 'play', 'behaviour', 'skills', 'baby', 'help']
...	...	...
105	150	['original', 'story', 'read', '', '', '', '', '', '', '']

Table 2.16: Topic Representation for the model with DBCV Score of 0.269222

2.3.6 Sixth Model

For our sixth model, we will utilize the entire dataset to validate the hyperparameters derived from our fifth model.

Our approach will commence with the removal of spam data previously identified in our models. Should it be required, we will undertake additional refinement of outlier topics to control their size effectively.

We conducted further preprocessing on the data, targeting rows containing phrases such as "original," "read," "comments or queries," "free breaking and community news," "www," "for the latest news," and "news your way download." Following this, we filtered out rows that appeared more than twice in the dataframe and removed instances where rows appeared twice from the same source. These actions aimed to reduce the presence of spam within the dataset. Given the extensive size of our dataset, it's impractical to eliminate every spam instance entirely. Consequently, we decided to revise our approach and remove duplicates from our data, which we had previously retained. Initially, we maintained duplicates to assign greater importance to data duplicated from different sources. However, in our efforts to thoroughly cleanse the data of spam and advertisements, we have opted to abandon this strategy and simply remove all duplicates.

Outlined below are the hyperparameters under consideration (see Table 2.17), accompanied by a column indicating potential adjustments.

n neighbors	n comp	min dist	metric	min cluster size	min samp	Adjustable Parameter
10	10	0.0	hamming	10	150	min cluster size
20	10	0.0	ghali	30	150	min cluster size
20	10	0.0	hamming	150	10	-

Table 2.17: Hyperparameters and Adjustable Parameters for Potential Optimization in the Final Model

We begin by examining the DBCV scores associated with each hyperparameter configuration (see Table 2.18), followed by an evaluation of the corresponding topics to validate their coherence (see Tables 2.19, 2.20, 2.21). Subsequently, we'll identify the best-performing hyperparameters and explore potential adjustments to further optimize our model.

n neighbors	n components	min dist	metric	min cluster size	min samples	DBCV
10	10	0.0	hamming	10	150	0.155939
20	10	0.0	ghali	30	150	0.154559
20	10	0.0	hamming	150	10	0.128192

Table 2.18: DBCV Score for the Hyperparameter sets Tested

Topic	Count	Representation
-1	672707	['said', 'people', 'water', 'government', 'police', 'south', 'public', 'court', 'state', 'national']
0	43005	['car', 'engine', 'cars', 'vehicle', 'rear', 'electric', 'vehicles', 'kw', 'models', 'ford']
1	35700	['cup', 'rugby', 'players', 'league', 'game', 'games', 'coach', 'match', 'cricket', 'football']
2	27819	['tax', 'dividend', 'bank', 'debt', 'investors', 'financial', 'banks', 'shares', 'ksh', 'inflation']
3	21275	['students', 'education', 'schools', 'school', 'teachers', 'learners', 'university', 'pupils', 'student', 'learning']
4	18327	['music', 'album', 'song', 'film', 'songs', 'story', 'musical', 'love', 'band', 'book']
5	17513	['suspects', 'shot', 'vehicle', 'robbery', 'robbers', 'police', 'armed', 'men', 'stolen', 'robbed']
6	14118	['anc', 'zuma', 'party', 'jacob', 'president', 'zumas', 'ramaphosa', 'ancs', 'conference', 'partys']
7	11104	['airline', 'airlines', 'aircraft', 'aviation', 'airport', 'flights', 'flight', 'boeing', 'plane', 'airways']
8	9751	['food', 'farmers', 'agricultural', 'agriculture', 'hunger', 'farming', 'maize', 'insecurity', 'crops', 'crop']
9	7462	['gaza', 'israel', 'israeli', 'hamas', 'palestinian', 'palestinians', 'ceasefire', 'israels', 'rafah', 'aid']
10	7283	['women', 'gender', 'womens', 'equality', 'female', 'empowerment', 'men', 'leadership', 'inclusion', 'girls']
11	6556	['rhino', 'poaching', 'rhinos', 'elephants', 'elephant', 'lion', 'lions', 'horn', 'wildlife', 'animals']
12	6519	['eskom', 'eskoms', 'loadshedding', 'power', 'utility', 'shedding', 'electricity', 'load', 'stage', 'mw']
13	5833	['hope', 'future', 'life', 'think', 'nation', 'im', 'better', 'let', 'things', 'dreams']
14	5788	['property', 'rental', 'buyers', 'tenants', 'estate', 'properties', 'home', 'rent', 'tenant', 'landlords']
15	5712	['mining', 'gold', 'mineral', 'miners', 'mines', 'platinum', 'minerals', 'metals', 'copper', 'ore']
16	5450	['waste', 'plastic', 'recycling', 'plastics', 'landfill', 'recycled', 'bags', 'pollution', 'packaging', 'dumping']
17	4932	['plants', 'flowers', 'plant', 'leaves', 'garden', 'soil', 'flower', 'grow', 'blooms', 'watering']
18	4635	['dog', 'dogs', 'pet', 'spca', 'pets', 'animal', 'animals', 'breed', 'puppy', 'cats']
19	4606	['hotel', 'mountain', 'hiking', 'spa', 'hotels', 'rooms', 'mountains', 'beach', 'guests', 'island']
...	...	...
482	10	['arts', 'cultural', 'artists', 'culture', 'caliber', 'attributed', 'egos', 'intimately', 'paves', 'creative']

Table 2.19: List of topics with their count and word representations for the first set of hyperparameters in our final model

Topic	Count	Representation
-1	738736	['said', 'people', 'police', 'government', 'south', 'public', 'africa', 'years', 'court', 'new']
0	50422	['cup', 'players', 'rugby', 'league', 'game', 'games', 'coach', 'match', 'football', 'win']
1	20840	['education', 'students', 'school', 'schools', 'learners', 'teachers', 'university', 'pupils', 'learning', 'student']
2	14745	['water', 'sanitation', 'dam', 'river', 'sewage', 'drinking', 'supply', 'dams', 'vaal', 'toilets']
3	14097	['suspects', 'shot', 'vehicle', 'robbery', 'robbers', 'police', 'armed', 'men', 'robbed', 'stolen']
4	13362	['rear', 'car', 'engine', 'kw', 'electric', 'models', 'model', 'inch', 'cars', 'wheels']
5	12866	['anc', 'zuma', 'party', 'jacob', 'president', 'ramaphosa', 'ancs', 'conference', 'zumas', 'partys']
6	11574	['add', 'sauce', 'butter', 'cheese', 'pepper', 'chicken', 'salt', 'cook', 'chopped', 'minutes']
7	11222	['airline', 'airlines', 'aircraft', 'airport', 'aviation', 'flights', 'flight', 'boeing', 'plane', 'airways']
8	9219	['farmers', 'food', 'agriculture', 'agricultural', 'farming', 'maize', 'hunger', 'crops', 'production', 'crop']
9	9088	['women', 'gender', 'womens', 'equality', 'female', 'empowerment', 'girls', 'men', 'leadership', 'inclusion']
10	7613	['climate', 'change', 'emissions', 'warming', 'carbon', 'global', 'greenhouse', 'paris', 'temperatures', 'cop']
11	7362	['shares', 'ksh', 'share', 'kshs', 'net', 'profit', 'investors', 'earnings', 'billion', 'rate']
12	7191	['mining', 'gold', 'miners', 'mineral', 'mines', 'minerals', 'platinum', 'copper', 'metals', 'ore']
13	6895	['gaza', 'israel', 'israeli', 'hamas', 'palestinian', 'palestinians', 'ceasefire', 'israels', 'rafah', 'aid']
14	6505	['music', 'album', 'song', 'songs', 'band', 'musical', 'artists', 'musicians', 'artist', 'jazz']
15	6434	['eskom', 'eskoms', 'loadshedding', 'power', 'utility', 'shedding', 'load', 'electricity', 'mw', 'stage']
16	5779	['energy', 'solar', 'renewable', 'power', 'electricity', 'wind', 'mw', 'grid', 'nuclear', 'generation']
17	5545	['waste', 'plastic', 'recycling', 'plastics', 'landfill', 'bags', 'management', 'recycled', 'pollution', 'dumping']
18	5346	['apartheid', 'racism', 'mandela', 'white', 'black', 'racial', 'africans', 'hate', 'south', 'racist']
19	5196	['unemployment', 'skills', 'youth', 'employees', 'employment', 'job', 'jobs', 'work', 'unemployed', 'training']
...	...	...
383	31	['insurance', 'insurers', 'headquartered', 'asset', 'renewals', 'swiss', 'franchise', 'subsector', 'zurich', 'growth']

Table 2.20: List of topics with their count and word representations for the second set of hyperparameters in our final model

Topic	Count	Representation
-1	737124	['said', 'people', 'police', 'government', 'south', 'public', 'years', 'court', 'africa', 'time']
0	49050	['cup', 'players', 'rugby', 'league', 'game', 'games', 'coach', 'match', 'football', 'team']
1	33057	['car', 'engine', 'cars', 'vehicle', 'vehicles', 'rear', 'electric', 'kw', 'models', 'model']
2	14686	['anc', 'zuma', 'party', 'jacob', 'president', 'zumas', 'ramaphosa', 'ancs', 'conference', 'partys']
3	14246	['education', 'students', 'schools', 'school', 'learners', 'teachers', 'university', 'student', 'learning', 'universities']
4	13652	['suspects', 'shot', 'vehicle', 'robbery', 'robbers', 'police', 'armed', 'men', 'scene', 'robbed']
5	13081	['music', 'album', 'song', 'songs', 'musical', 'band', 'artists', 'love', 'film', 'comedy']
6	10850	['airline', 'airlines', 'aircraft', 'aviation', 'airport', 'flights', 'flight', 'boeing', 'plane', 'airways']
7	8876	['add', 'sauce', 'pepper', 'butter', 'chicken', 'salt', 'chopped', 'heat', 'minutes', 'cook']
8	8464	['race', 'racing', 'class', 'championship', 'formula', 'bike', 'rally', 'races', 'overall', 'max']
9	8349	['water', 'dam', 'sanitation', 'supply', 'river', 'sewage', 'vaal', 'drinking', 'dams', 'residents']
10	8164	['women', 'gender', 'womens', 'equality', 'female', 'empowerment', 'men', 'girls', 'leadership', 'inclusion']
11	7472	['gaza', 'israel', 'israeli', 'hamas', 'palestinian', 'ceasefire', 'palestinians', 'israels', 'rafah', 'aid']
12	5771	['climate', 'change', 'warming', 'emissions', 'global', 'paris', 'temperatures', 'greenhouse', 'ice', 'carbon']
13	5680	['food', 'farmers', 'agriculture', 'agricultural', 'hunger', 'maize', 'farming', 'insecurity', 'production', 'crop']
14	5331	['rhino', 'poaching', 'lion', 'lions', 'elephants', 'elephant', 'horn', 'horns', 'wildlife', 'leopard']
15	5080	['waste', 'plastic', 'recycling', 'plastics', 'landfill', 'bags', 'recycled', 'management', 'pollution', 'dumping']
16	4819	['cyber', 'cybersecurity', 'security', 'hackers', 'data', 'information', 'attacks', 'passwords', 'password', 'fraud']
17	4701	['mining', 'gold', 'mineral', 'miners', 'platinum', 'minerals', 'mines', 'copper', 'metals', 'exploration']
18	4065	['property', 'rental', 'buyers', 'tenants', 'properties', 'estate', 'home', 'tenant', 'agent', 'rent']
19	3986	['art', 'artists', 'auction', 'strauss', 'exhibition', 'artist', 'works', 'contemporary', 'painting', 'museum']
...	...	...
372	150	['highlights', 'look', 'signs', 'know', 'following', 'heres', 'include', 'suggestions', 'things', 'looking']

Table 2.21: List of topics with their count and word representations for the third set of hyperparameters in our final model

Upon reviewing the DBCV scores, they appear slightly lower compared to their counterparts from the 200,000-documents models. This discrepancy may be attributed to the removal of a substantial portion, if not all, of the spam/ads that the models would have otherwise categorized into topics, thereby reducing the number of documents associated with each topic. Upon initial inspection of the topic representations, we observe the absence of spam-related topics, and the topics themselves appear to convey meaningful content.

The `min_cluster_size` parameter in HDBSCAN determines the minimum number of neighboring points required for a point to be classified as a core point. Lowering this parameter is likely to yield improved results, albeit to a limited extent. Since this is the only adjustable hyperparameter we've derived from the fifth model, our initial focus will be on validating the hyperparameters first then enhancing it furthermore by reducing the outliers before maybe fine-tuning the `min_cluster_size` to enhance our model further.

Our next step to validate the models involves assessing the similarity of topics to ensure sufficient diversity. This is achieved by examining the cosine similarity heatmap (see Figure).

We observe considerable similarity among the topics across the three models. We will investigate whether this similarity arises due to the calculation of cosine similarities using quantized embeddings.

We will proceed to evaluate the cosine similarities of the topics utilizing the original embeddings.

Regrettably, this yields identical results. Therefore, to discern whether this outcome is attributable to the application of quantized embeddings during dimensionality reduction and clustering, or if it's due to the inherent similarity of our data derived from an ESG framework, we will attempt hyperparameter tuning using the original embeddings.

2.3.7 Seventh Model

For this model, we will search for the optimal hyperparameters using the original embeddings. We will employ a dataset of 400k randomly sampled entries, ensuring replicable results through the use of a pandas random state.

Given that we are solely utilizing the "cosine" metric and recognizing that certain hyperparameters have minimal impact on model performance, we will test a reduced set of hyperparameters. To expedite the search process, we will maintain some parameters at their default values: `n_components=5`. Specifically, we will explore the following hyperparameters: `n_neighbors=(5, 10, 20, 50, 100)`, `min_dist=(0.0, 0.1, 0.25)`, `min_cluster_size=(10, 15, 30, 60, 150)`, and `min_samples=(10, 15, 30, 60, 150)`.

Initially, we will examine the models exhibiting a DBCV score exceeding 0.3. Referring to Table 2.22, it's evident that only one set of hyperparameters forms a group meeting this criterion, accompanied by a solitary outlier. This group comprises `n_neighbors=50, 100`,

`min_dist=0.25`, `min_cluster_size=10, 15, 30`, and `min_samples=10, 15, 30`, while the outlier is characterized by `n_neighbors=5`, `min_dist=0.25`, `min_cluster_size=150`, and `min_samples=15`.

n neighbors	min dist	min cluster size	min samples	DBCV
50	0.25	10	10	0.835273
50	0.25	15	10	0.835273
50	0.25	30	10	0.835273
50	0.25	15	15	0.833373
50	0.25	10	15	0.833373
50	0.25	30	15	0.833373
50	0.25	30	30	0.824103
50	0.25	15	30	0.824103
50	0.25	10	30	0.824103
100	0.25	10	10	0.751127
100	0.25	30	10	0.751127
100	0.25	15	10	0.751127
100	0.25	15	15	0.747377
100	0.25	10	15	0.747377
100	0.25	30	15	0.747377
100	0.25	15	30	0.737217
100	0.25	30	30	0.737217
100	0.25	10	30	0.737217
5	0.25	150	15	0.498047

Table 2.22: Hyperparameter Sets for DBCV Exceeding 0.3

For a more accurate observation, we will examine models separately for `n_neighbors=50` and `n_neighbors=100`.

For `n_neighbors=50` models, we will focus on the model characterized by `min_dist=0.25`, `min_cluster_size=10`, and `min_samples=10`. Despite its impressive 0.835273 DBCV score, similar results were encountered in the previous hyperparameter tuning process with poor representations. Therefore, our initial investigation will be directed towards its topic representation.

Topic	Count	Representation
0	399955	['said', 'south', 'people', 'new', 'africa', 'year', 'time', 'years', 'government', 'police']
1	45	['shared', 'post', 'aimeekitshoff', 'rugbypass', 'lasizwe', 'bryoni', 'ingrid', 'magazine', 'majorleaguedjz', 'mscosmosa']

Table 2.23: List of topics with their count and word representations for the hyperparameter set with a DBCV score of 0.835273

Observing Table 2.23, we note that the topic representation is limited to only 2 topics, with Topic 0 containing 99.9% of the documents, indicating significant imbalance. While

this outcome aligns with our expectations based on previous hyperparameter tuning processes, it underscores the unsuitability of these hyperparameters for our objectives.

Next, we will examine the model characterized by `n_neighbors=100`, `min_dist=0.25`, `min_cluster_size=10`, and `min_samples=10`. Consistent with previous evaluations, our initial focus will be on its topic representation.

Topic	Count	Representation
0	399955	['said', 'south', 'people', 'new', 'africa', 'year', 'time', 'years', 'government', 'police']
1	45	['shared', 'post', 'aimeekitshoff', 'rugbypass', 'lasizwe', 'bryoni', 'ingrid', 'magazine', 'majorleaguedjz', 'mscosmosa']

Table 2.24: List of topics with their count and word representations for the hyperparameter set with a DBCV score of 0.751127

Observing Table 2.24, we notice that the topic representation is imbalanced, comprising only 2 topics, with the majority of the data concentrated in the first topic. This imbalance renders the representation highly unsuitable for our use case. Consequently, we will refrain from utilizing the "blue" and "red" sets of hyperparameters from Table 2.22, as they do not significantly differ from the two sets we have already examined, thus likely yielding similarly poor results.

Finally, we will examine the last hyperparameter set with a DBCV score exceeding 0.3. This model is characterized by `n_neighbors=5`, `min_dist=0.25`, `min_cluster_size=150`, and `min_samples=15`, with a DBCV score of 0.498047.

The absence of any similar hyperparameter sets with comparable results—given that this model achieves a DBCV score of nearly 0.5, and no other model with similar hyperparameters surpasses 0.3—strongly suggests the potential inadequacy of this model. Nonetheless, we will scrutinize its representation to confirm our suspicion.

Topic	Count	Representation
-1	1626	['market', 'health', 'old', 'sleep', 'corruption', 'government', 'moyo', 'tfp', 'said', 'growth']
0	381434	['said', 'people', 'south', 'new', 'africa', 'year', 'time', 'years', 'government', 'police']
1	16231	['cup', 'world', 'players', 'team', 'game', 'league', 'rugby', 'said', 'games', 'coach']
2	314	['haiti', 'haitian', 'haitis', 'henry', 'gangs', 'gang', 'mission', 'portauprince', 'prime', 'haitians']
3	235	['airport', 'crossword', 'philippines', 'code', 'dialing', 'clue', 'airports', 'mizan', 'teferi', 'segou']
4	160	['race', 'horse', 'horses', 'stakes', 'racing', 'jockey', 'handicap', 'fillies', 'time', 'trainer']

Table 2.25: List of topics with their count and word representations for the hyperparameter set with a DBCV score of 0.498047

Examining Table 2.25, which illustrates the list of topics for the model with a DBCV score of 0.49, we observe that the majority of the data is assigned to topic 0, lacking meaningful representation and comprising over 95% of the data. Consequently, this representation also proves poorly suitable for our use case.

We can confidently conclude that the hyperparameter sets resulting in a DBCV score of more than 0.3 (see Table 2.22) are all inadequate for our use case. Therefore, our focus will now shift to those sets with scores between 0.25 and 0.3.

n neighbors	min dist	min cluster size	min samples	DBCV
20	0.0	60	150	0.294759
20	0.0	30	150	0.293583
20	0.0	10	150	0.293035
20	0.0	15	150	0.283582
20	0.0	150	150	0.282546
5	0.0	150	10	0.271128
5	0.0	150	30	0.269721
50	0.0	150	30	0.269460
50	0.0	150	150	0.262184
20	0.0	150	30	0.257853
5	0.0	150	60	0.257146
5	0.0	60	60	0.256789
50	0.0	150	60	0.255595
100	0.0	60	30	0.253633
100	0.0	150	15	0.253140

Table 2.26: Hyperparameter Sets for DBCV Between 0.25 and 0.3

We assign the same color to hyperparameter sets that differ from each other by only one hyperparameter (see Table 2.26). Typically, these sets yield results that align closely with each other. Thus, observing one set with a particular color is akin to observing every single set sharing that color. Consequently, we will focus on examining the best-performing set from each color group, serving as a representative of the entire group.

Our evaluation process entails first scrutinizing the topic representation to assess whether the topics are sufficiently refined and detailed for our use case. Subsequently, if the representation meets our criteria, we will analyze the similarity matrix between the topics to ensure their diversity and differentiation, ensuring that the clustering algorithm effectively delineated between the topics. This comprehensive analysis aids in clustering the remaining data into relevant topics, facilitating subsequent sentiment analysis.

So first, let's take a look at our first model which consists of `n_neighbors=20`, `min_dist=0.0`, `min_cluster_size=60`, and `min_samples=150`.

Topic	Count	Representation
-1	214715	['said', 'people', 'police', 'government', 'south', 'public', 'africa', 'years', 'time', 'court']
0	16630	['cup', 'players', 'rugby', 'game', 'league', 'games', 'coach', 'team', 'football', 'match']
1	10486	['car', 'engine', 'vehicle', 'cars', 'vehicles', 'rear', 'electric', 'kw', 'model', 'models']
2	7108	['education', 'students', 'school', 'schools', 'learners', 'teachers', 'university', 'learning', 'pupils', 'student']
3	4939	['water', 'sanitation', 'dam', 'river', 'supply', 'toilets', 'drinking', 'sewage', 'residents', 'vaal']
4	4459	['shares', 'dividend', 'ksh', 'shareholders', 'share', 'bank', 'billion', 'investors', 'stock', 'banks']
5	3877	['airline', 'airport', 'aircraft', 'airlines', 'aviation', 'flight', 'flights', 'boeing', 'air', 'plane']
6	3837	['add', 'sauce', 'salt', 'pepper', 'chicken', 'cook', 'butter', 'chopped', 'dish', 'minutes']
7	3430	['heritage', 'hotel', 'mountain', 'ancient', 'park', 'beach', 'travel', 'cultural', 'guests', 'hiking']
8	3108	['mining', 'gold', 'mines', 'miners', 'mineral', 'platinum', 'minerals', 'metals', 'lithium', 'ore']
9	2807	['rhino', 'poaching', 'elephants', 'lions', 'elephant', 'rhinos', 'lion', 'animals', 'wildlife', 'hunting']
10	2741	['gaza', 'israel', 'israeli', 'hamas', 'ceasefire', 'palestinian', 'palestinians', 'israels', 'aid', 'rafah']
11	2638	['music', 'album', 'songs', 'song', 'band', 'artists', 'musical', 'jazz', 'singer', 'dj']
12	2609	['apartheid', 'mandela', 'black', 'racism', 'south', 'white', 'racial', 'africans', 'freedom', 'nelson']
13	2581	['anc', 'zuma', 'party', 'president', 'conference', 'ramaphosa', 'ancs', 'nec', 'jacob', 'dlaminizuma']
14	2559	['women', 'gender', 'womens', 'equality', 'female', 'empowerment', 'men', 'leadership', 'girls', 'woman']
15	2398	['workers', 'sassa', 'strike', 'grant', 'grants', 'union', 'employees', 'labour', 'unions', 'wage']
16	2398	['court', 'judge', 'justice', 'matter', 'legal', 'comment', 'judges', 'appeal', 'constitutional', 'jsc']
17	2222	['child', 'children', 'parents', 'kids', 'childs', 'baby', 'sleep', 'play', 'help', 'fun']
18	2201	['health', 'hospital', 'healthcare', 'hospitals', 'medical', 'nurses', 'patients', 'nhi', 'doctors', 'care']
19	2150	['data', 'cyber', 'security', 'information', 'cybersecurity', 'privacy', 'personal', 'hackers', 'identity', 'attacks']
...	...	...
199	60	['virus', 'pandemic', 'measures', 'spread', 'continue', 'covid', 'wave', 'countries', 'curve', 'vigilant']

Table 2.27: List of topics with their count and word representations for the hyperparameter set with a DBCV score of 0.294759

Examining Table 2.27, we observe that the model exhibits sufficient granularity and coherent topics. Therefore, we proceed to evaluate the similarity matrix, which depicts the cosine similarity scores between the topics.

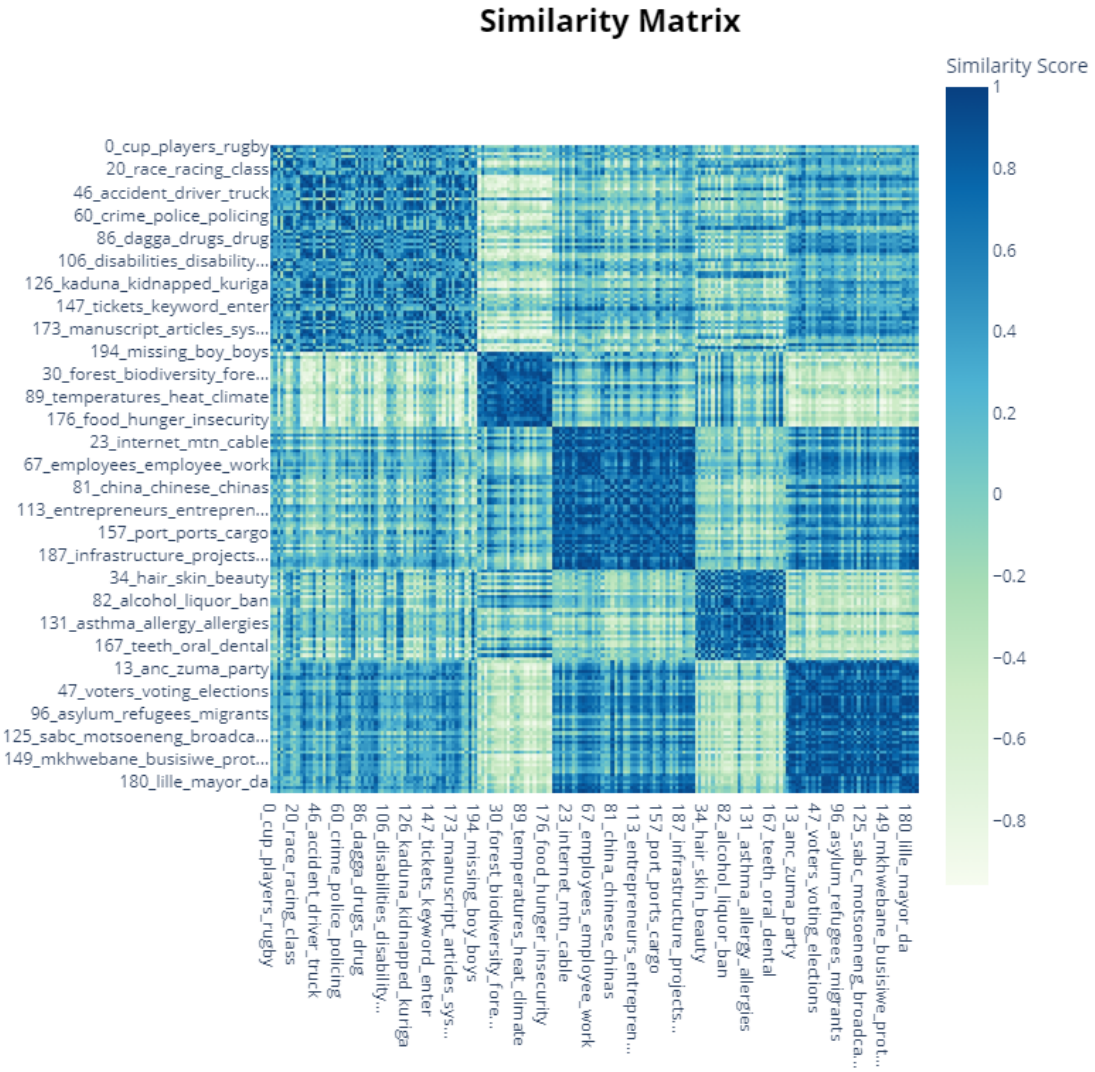


Fig. 2.9: Similarity Matrix for the topics of the model with 0.294759 DBCV score

In Figure 2.9, we note that although there is still noticeable similarity between the topics, the similarity scores range between -1 and 1. This variability ensures that cosine similarity can effectively allocate future documents and facilitate subsequent sentiment analysis.

Next, we'll examine our second set of hyperparameters, characterized by `n_neighbors=5`, `min_dist=0.0`, `min_cluster_size=150`, and `min_samples=10`.

Given that topic representation typically isn't problematic within this range of DBCV scores, we'll initially examine the similarity matrix. Observing Figure 2.10, we observe that all topics exhibit significant correlation, with similarity scores ranging from 0.5 to

1. Most topics are highly similar, with scores predominantly falling within the 0.9 to 1 range. This high level of similarity complicates the task of allocating future documents and performing sentiment analysis, rendering this set of hyperparameters unsuitable for our use case.

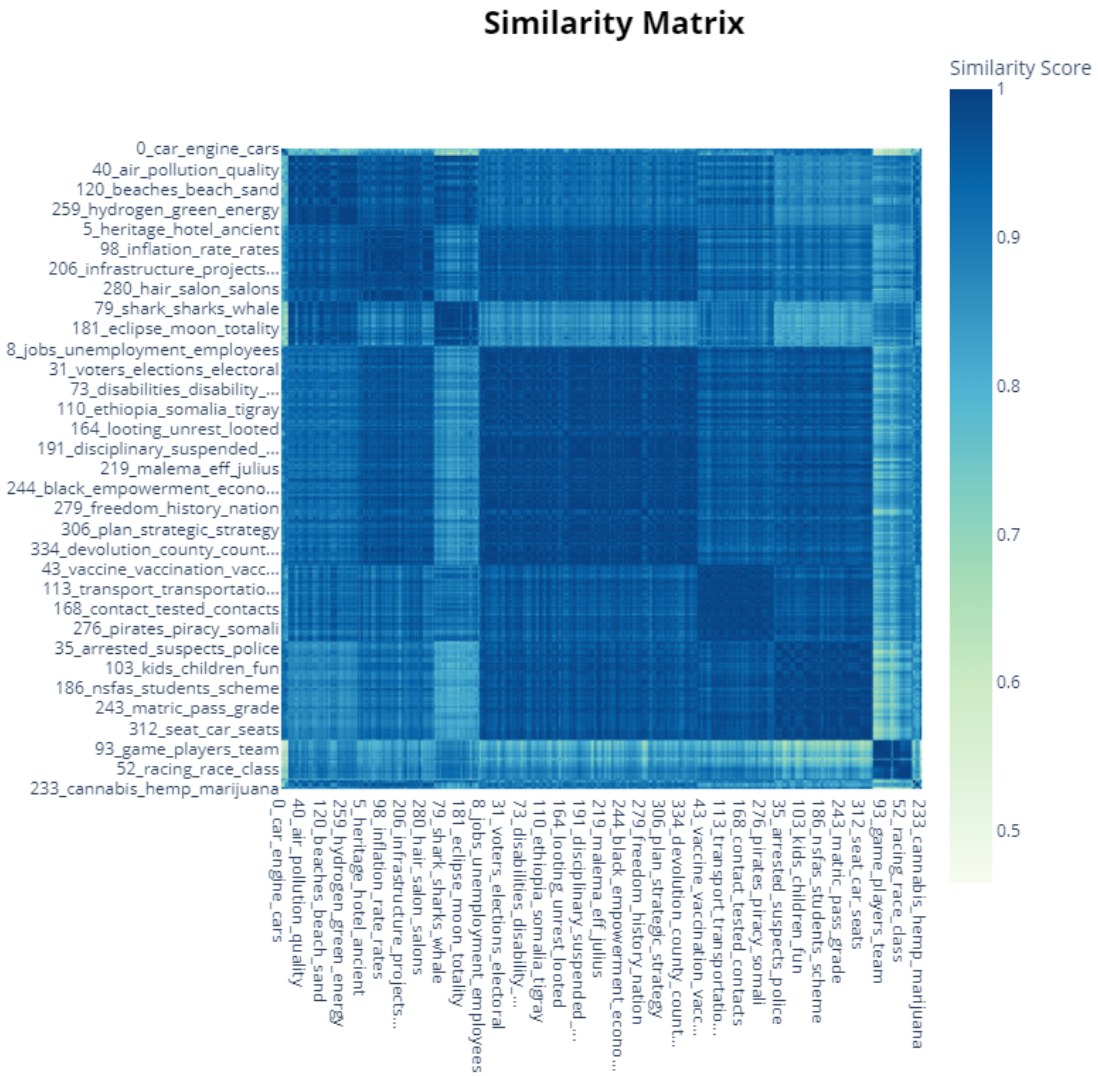


Fig. 2.10: Similarity Matrix for the topics of the model with 0.271128 DBCV score

The next set of hyperparameters we'll be examining is `n_neighbors=50`, `min_dist=0.0`, `min_cluster_size=150`, and `min_samples=30`.

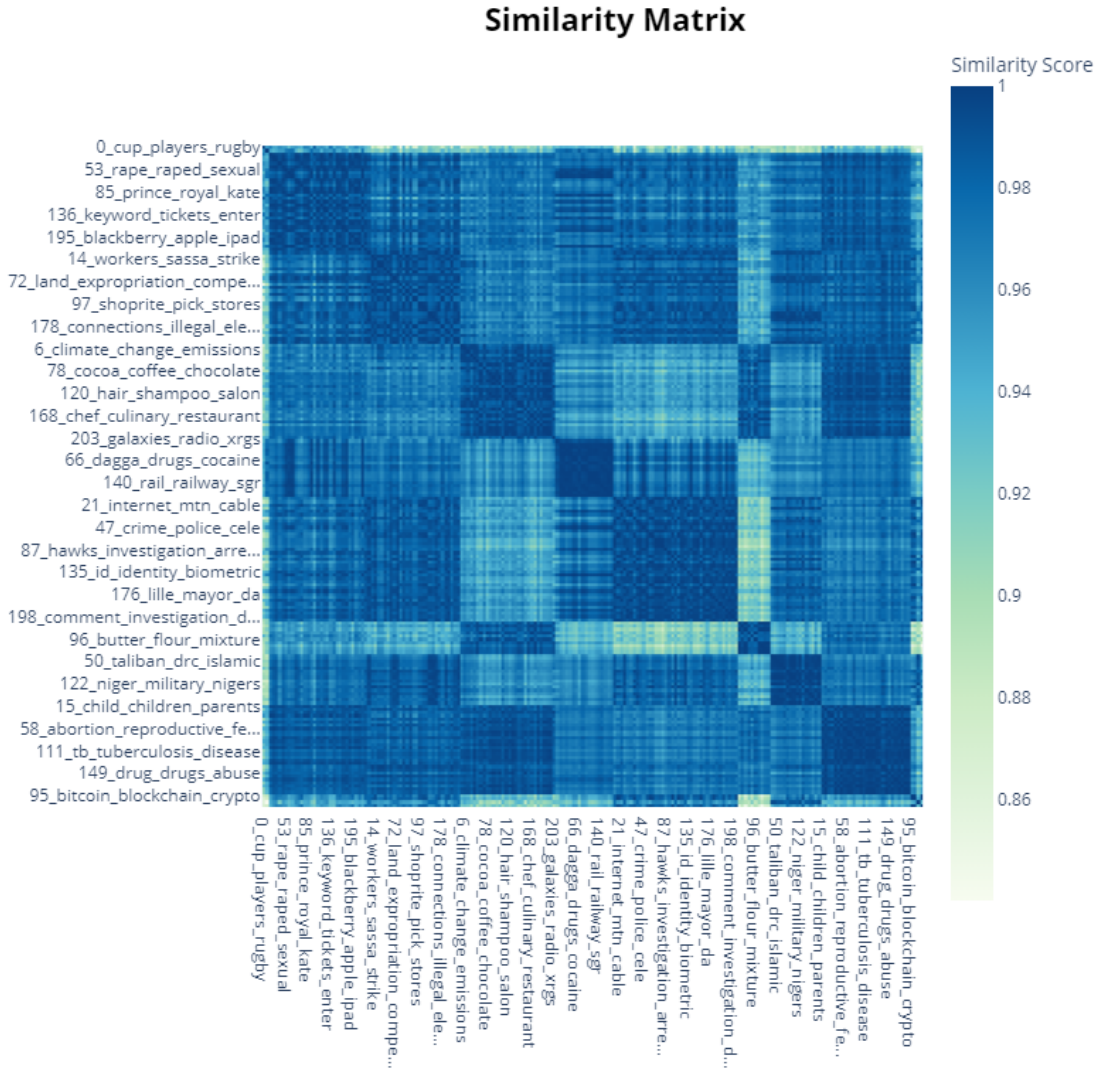


Fig. 2.11: Similarity Matrix for the topics of the model with 0.269460 DBCV score

Examining Figure 2.11, we notice significant correlation among all topics, with similarity scores ranging from 0.84 to 1. Most topics demonstrate high similarity, with scores primarily within the 0.9 to 1 range. Utilizing the same rationale as our previous assessment, we find this set of hyperparameters unsuitable for our use case.

The subsequent set of hyperparameters is `n_neighbors=100`, `min_dist=0.0`, `min_cluster_size=60`, and `min_samples=30`.

Upon reviewing Figure 2.12, it becomes evident that there is notable correlation among all topics, indicated by similarity scores ranging from 0.65 to 1. Most topics exhibit high similarity, with scores predominantly falling within the 0.9 to 1 range. Employing the same reasoning as in our prior evaluations, we conclude that this set of hyperparameters is unsuitable for our use case.

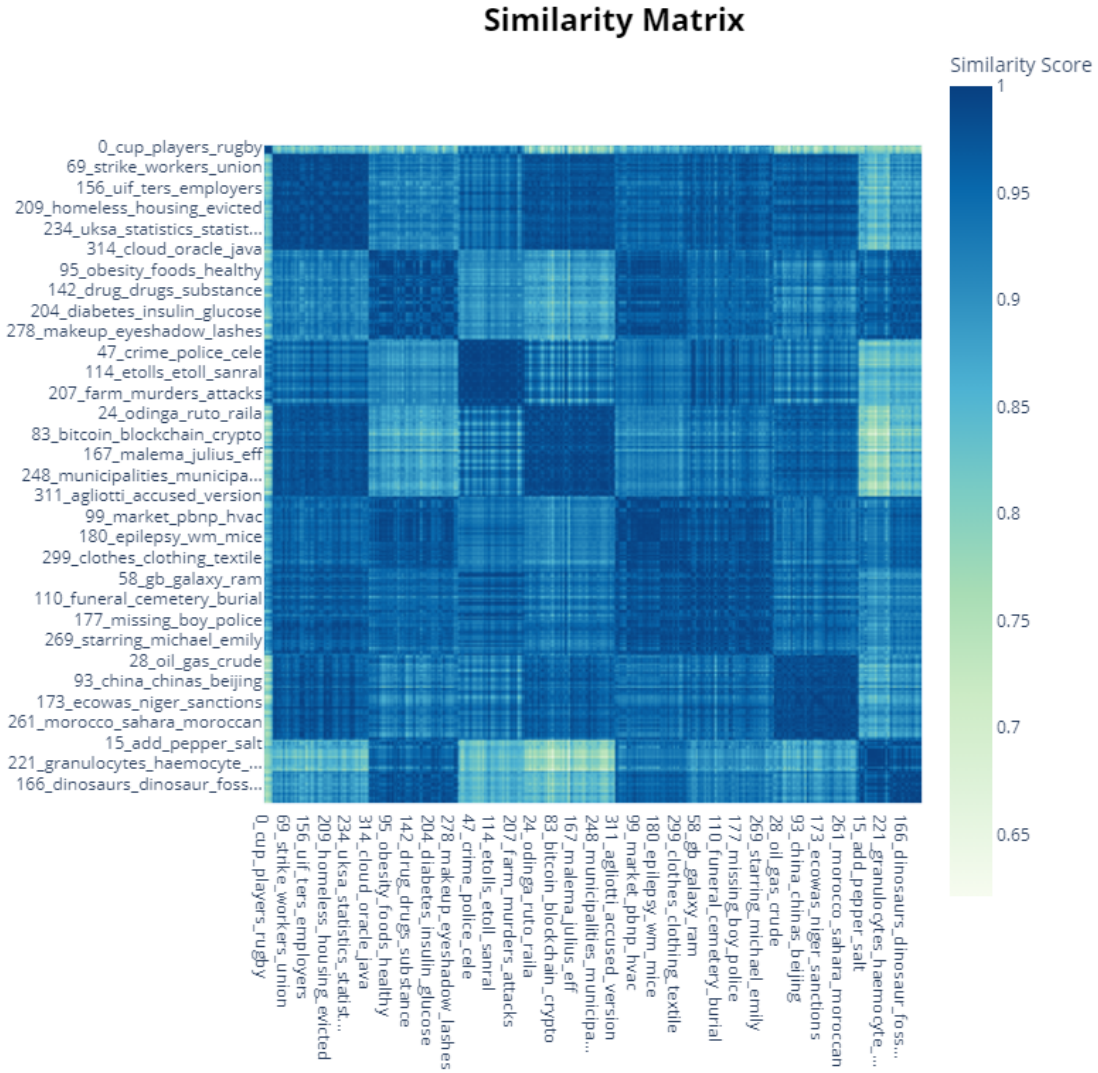


Fig. 2.12: Similarity Matrix for the topics of the model with 0.253633 DBCV score

Finally, we have `n_neighbors=100`, `min_dist=0.0`, `min_cluster_size=150`, and `min_samples=15`.

After examining Figure 2.13, it's clear that there's significant correlation among all topics, as evidenced by similarity scores ranging from 0.65 to 1. The majority of topics display high similarity, with scores mostly clustered within the 0.9 to 1 range. Applying the same logic as in our previous assessments, we determine that this final set of hyperparameters is likewise unfit for our use case.

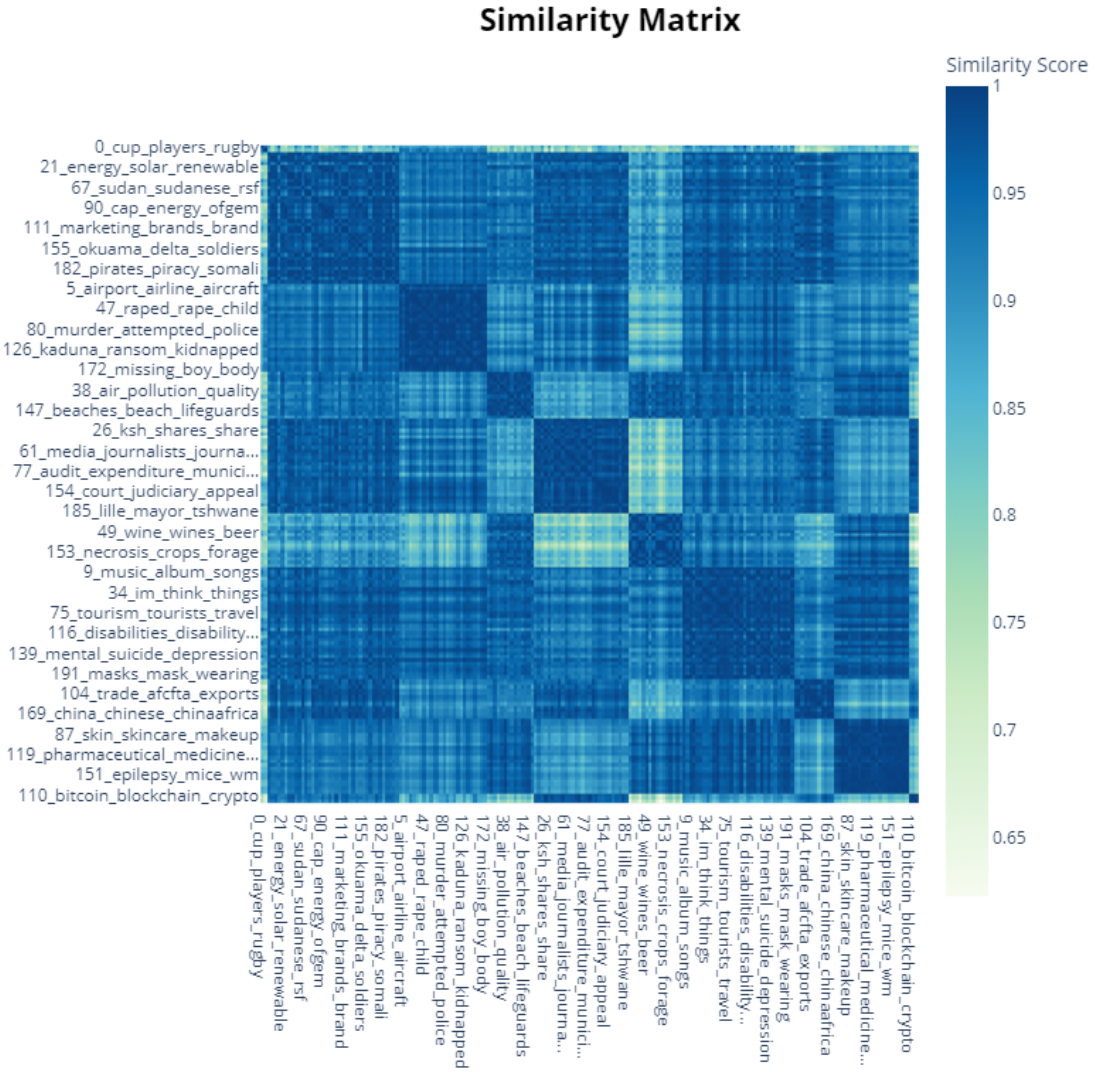


Fig. 2.13: Similarity Matrix for the topics of the model with 0.253140 DBCV score

The only set of hyperparameters that proved suitable for our use case among those we examined is `n_neighbors=20`, `min_dist=0.0`, `min_cluster_size=60`, and `min_samples=150`. This set not only yields the best DBCV score (excluding models with poor topic representation), but it also demonstrates good topic representation and granularity. Moreover, the topics exhibit diversity, as indicated by the cosine similarity scores ranging from -1 to 1. Therefore, we will consider using this set of hyperparameters for our final model.

2.3.8 Final Model Selection and Challenges in Incorporating Additional Data

To incorporate the additional data into our model, we must generate embeddings that align with the dimensionality reduction process. However, a significant challenge arises with UMAP due to its tendency to yield inconsistent outcomes even when hyperparameters

remain constant. This inconsistency arises because UMAP produces comparable results only after undergoing a Procrustes transformation, despite utilizing the same data and hyperparameters. In an attempt to leverage the entirety of our data, we explored the utilization of Aligned-UMAP.

Aligned-UMAP operates similarly to conventional UMAP, reducing the dimensionality of data across multiple partitions. However, it distinguishes itself by aligning outcomes through shared points across these partitions. Regrettably, employing Aligned-UMAP necessitates maintaining overlapping slices of data in memory, which significantly increases memory usage compared to using the entire dataset. Additionally, aligning each pair of slices requires a minimum of 2 hours. Considering a 250,000 overlap between every two slices, aligning 7 slices renders the task impractical.

We also experimented with employing a simple neural network to train it on our 400k reduced dataset, aiming to teach it how to reduce embeddings. However, the results were suboptimal across various architectures.

Consequently, we opt to adhere to the data utilized in the previous model, as it represents a random sample comprising one-third of the entire dataset, making it a representative subset. We will employ this subset for sentiment analysis.

Now, we can utilize this model to examine the evolution of topics over time. This involves treating each topic as a class that requires examination at various time intervals, thereby generating a distinct topic representation for each timestep. Through this approach, we gain insight into the evolution of the prominent terms associated with each topic at different points in time. Moreover, it aids in validating our model and offers finer-grained insights into the dataset.

2.4 Results of Sentiment Analysis

The goal of the sentiment analysis is to glean deeper insights from our dataset. We will be utilizing our last model to try to get score relaying how good the sentiment in regards to a specific keyword from our ESG and SDGs frameworks.

2.4.1 Preliminary Step

Now that our topic model is constructed, a crucial way to get deeper insights involves conducting sentiment analysis. This process entails associating each keyword from our ESG and SDG framework with the topics derived from our topic modeling phase, achieved through cosine similarity computation between the reduced embeddings of the framework's keywords and the embeddings of the topics. These topic embeddings are the centroids derived from the reduced embeddings of documents within each topic.

To accomplish this, we must obtain embeddings for every individual keyword and then employ Aligned-UMAP to align the newly reduced embeddings with the existing dataset.

This approach avoids generating fresh reduced embeddings with the new keywords as additional documents, preserving the integrity of our established model while ensuring alignment with our topic modeling framework. Table 2.28 shows the results of Aligned-UMAP.

keywords	embeddings	reduced embeddings
Health hazards	[-0.01059488 -0.0212846 -0.03541314 ... -0.01120811 0.00632847 0.0112278]	[0.45100638 -0.43243381 -0.47580793 2.06632543 0.43698564]
death	[0.01522458 -0.00505549 -0.03483903 ... -0.01107242 0.02643435 0.00970025]	[1.26785994 0.11597852 -0.62436593 -0.85321695 0.71385014]
Disability	[9.23286937e-03 1.21219293e-03 -4.37428467e-02 ... 1.13599794e-02 3.66601758e-02 -7.34170389e-05]	[1.42890465 -0.45069739 -1.25056362 -0.25608549 -0.92152429]
Physical activity	[0.0194313 0.00903889 -0.04539069 ... -0.01791616 0.03126863 -0.00425057]	[1.73047245 -0.36443079 -0.13756754 0.59579742 -0.81599927]
Housing affordability	[-0.00287061 0.00490231 -0.03135166 ... -0.00238879 0.03480121 0.01376239]	[-1.12233472 -0.69949728 -0.63595569 0.41416645 0.14299376]
Housing availability	[0.00239635 0.01716672 -0.03733461 ... -0.00283104 0.03793707 -0.0004112]	[-1.01244283 -0.69314843 -0.58340383 0.35350233 0.27915987]
Housing quality	[0.01128193 0.00988665 -0.0465121 ... -0.00921187 0.02934796 0.00678701]	[-0.9889968 -0.6392467 -0.61956787 0.31525582 0.31427974]
City attractiveness	[0.00987043 -0.00525569 -0.04096377 ... 0.00346693 0.0335347 0.01263461]	[-0.57813281 0.07879752 -0.25450897 1.02300966 -0.15098637]
Cultural diversity	[0.00661658 0.00117216 -0.03054632 ... 0.01062938 0.02778853 0.02512045]	[0.20916884 0.42428562 -1.01107967 -1.00409222 -0.93124807]
Cultural heritage protection	[-0.00135948 0.02450876 -0.00997565 ... -0.00175994 0.03816665 -0.0024291]	[0.21081987 1.28426564 -0.21795446 0.49325398 -0.34497666]
Education affordability	[-0.00370588 0.00556282 -0.03118945 ... 0.00345047 0.02221121 0.02924787]	[0.32976517 -1.26592624 -1.40337074 -1.17933738 -1.1389848]
...	...	...
Global partnership Sustainable Developement	[0.031868 -0.00277711 -0.04922682 ... -0.00080616 0.00517502 0.01626203]	[-1.32049584 0.37469387 -0.36980006 1.44321167 -0.03393303]

Table 2.28: Mapping ESG and SDGs Keywords: Original and Aligned Reduced Embeddings

With reference to Table 2.28, we can employ it to identify the topics with the highest similarity scores for each keyword. This can be achieved by either setting a threshold on

the similarity score or selecting the top n topics.

We will use 'bert-base-multilingual-uncased-sentiment,' a BERT-based sentiment analysis model, to determine the sentiment of each document. This model supports multiple languages, including English, Dutch, German, French, Spanish, and Italian, making it suitable for our use case with English documents. However, it is limited to 512 tokens, which means it is better suited for analyzing single sentences rather than entire paragraphs. Since our documents are paragraphs due to our topic modeling step, we will feed the paragraphs directly to the model if they contain fewer than 512 tokens. If a paragraph exceeds this limit, we will analyze it sentence by sentence and then use the median sentiment score to determine the overall sentiment of the paragraph. This model predicts the sentiment of the document as a number between 1 and 5.

2.4.2 Sentiment Analysis for Specific Group

To gain valuable insights from our data, we will use specific groups within the ESG framework as keywords (see Table 2.29). For each keyword, we will identify the top 5 most similar topics and gather all the documents associated with these topics. We will then calculate the median sentiment score of these documents to represent the sentiment of each keyword.

The sentiment scores attributed to each keyword serve as indicators of the prevailing attitudes surrounding different topics. A score of 1, denoting a negative sentiment, may suggest that the media coverage depicts issues such as "Justice" and "Transparency and accountability" (see Table 2.30) in a critical manner, possibly emphasizing deficiencies or controversies within these realms. Conversely, a score of 4, representing a positive sentiment, might suggest that subjects like "Indoor air pollution" and "Outdoor air pollution" (see Table 2.30) are depicted as effectively managed or successfully addressed by authorities or initiatives, thus receiving favorable coverage in the press. These sentiment scores offer insights into how these topics are framed and discussed within media discourse, thereby providing valuable perspectives on public discourse and opinion formation pertaining to diverse societal issues.

IMPACT	SPECIFIC GROUP
Air quality	Indoor air pollution, Outdoor air pollution, Olfactory pollution
Attractiveness	City attractiveness
Biodiversity	Biodiversity protection, Ecosystem services, Biological diversity, Species population
Climate Change	Greenhouse gas emission
Cultural richness and heritage	Cultural diversity, Cultural heritage protection
Economic innovation, dynamism and competitiveness	Innovation, Local sector development, New industries
Education	Education affordability, Education availability, Education quality
Employment	Employment figures, Earnings quality, Quality of the working environment, Job security, Job safety
Environmental and health awareness and behaviour	Waste education, Energy use education, Water use education, Hygiene and sanitation education, Travel behaviour, Sex education, Food consumption and diet education, Economic production, Labour productivity
Good governance	Inclusivity, Justice, Local democracy, Evidence-based policy-making, Transparency and accountability
Housing	Housing affordability
Light pollution	Sky glow, Light intrusion, Glare
Mental health	Physical activity
Noise	Indoor noise, Outdoor noise
Peace and security	Housing quality
Physical health	Health hazards, death, Disability
Private wealth	Value of assets, Economic empowerment
Public budget	Available municipal budget, Available regional, State budget, Tax revenue, Non-tax revenue
Social participation	Civic participation, Community cohesion
Soil quality	Soil pollution, Soil degradation, Soil texture
Sustainable production and consumption	Natural resources depletion, Waste production, Waste management
Temperature	Indoor air temperature, Outdoor air temperature
Water quality	Water pollution, Water , salinisation, acidification, Water temperature, Water treatment
Work-life balance	Housing availability

Table 2.29: Impacts and their specific groups

SPECIFIC GROUP	TOPICS	SENTIMENT SCORE
Health hazards	[158 105 153 189 44]	3
death	[66 181 182 45 71]	3
Disability	[129 63 109 163 156]	4
Physical activity	[167 129 131 146 156]	3
Housing affordability	[8 197 69 178 64]	3

Housing availability	[8 197 64 69 184]	3
Housing quality	[8 197 64 184 196]	3
City attractiveness	[176 33 143 40 80]	3
Cultural diversity	[37 107 9 119 63]	3
Cultural heritage protection	[61 9 183 37 136]	3
Education affordability	[49 109 63 163 38]	3
Education availability	[49 109 63 163 129]	3
Education quality	[49 109 63 163 129]	3
Waste education	[176 33 77 165 143]	3
Energy use education	[143 69 8 114 80]	3
Water use education	[65 137 113 69 197]	3
Hygiene and sanitation education	[189 38 199 49 163]	3
Travel behaviour	[87 78 91 41 177]	4
Sex education	[63 129 109 188 156]	4
Food consumption and diet education	[158 44 105 163 6]	4
Civic participation	[119 179 10 26 155]	2
Community cohesion	[119 179 154 180 13]	2
Inclusivity	[29 3 55 49 37]	3
Justice	[179 126 90 36 169]	1
Local democracy	[179 117 155 150 119]	2
Evidence-based policy-making	[179 90 55 119 49]	2
Transparency and accountability	[174 180 92 72 179]	1
Economic production	[178 79 8 101 170]	3
Labour productivity	[49 3 178 79 8]	3
Employment figures	[49 3 178 79 8]	3
Earnings quality	[178 79 101 88 124]	2
Quality of the working environment	[49 38 163 199 189]	3
Job security	[55 152 86 122 112]	1
Job safety	[49 120 199 189 153]	2
Innovation	[3 29 89 56 101]	3
Local sector development	[178 79 101 8 170]	3
New industries	[79 178 3 101 170]	3
Value of assets	[178 79 101 88 124]	2
Economic empowerment	[119 49 178 3 79]	3
Available municipal budget	[174 184 180 72 76]	2
Available regional	[142 29 3 170 79]	3
State budget	[178 174 79 184 8]	2
Tax revenue	[178 79 101 8 124]	2
Non-tax revenue	[178 79 101 8 124]	2
Natural resources depletion	[176 33 140 69 137]	3
Waste production	[176 33 77 165 143]	3
Waste management	[176 165 33 77 143]	3
Biodiversity protection	[176 158 77 33 175]	3
Ecosystem services	[176 158 33 77 175]	3
Biological diversity	[158 176 77 175 33]	3
Species population	[158 176 77 6 175]	4
Greenhouse gas emission	[176 33 143 140 77]	3
Indoor air pollution	[165 113 134 74 151]	4
Outdoor air pollution	[165 113 134 74 151]	4

Olfactory pollution	[158 105 77 175 67]	3
Indoor noise	[146 83 34 167 17]	4
Outdoor noise	[146 83 17 34 167]	4
Soil pollution	[176 33 158 77 175]	3
Soil degradation	[176 33 158 77 175]	3
Soil texture	[176 33 158 77 175]	3
Sky glow	[7 61 52 58 176]	4
Light intrusion	[31 175 77 198 51]	4
Glare	[175 77 31 198 51]	4
Water pollution	[65 113 137 69 197]	3
Water	[113 65 137 69 197]	3
salinisation	[176 33 158 77 175]	3
acidification	[176 158 33 77 175]	3
Water temperature	[113 65 137 69 165]	3
Water treatment	[65 113 137 69 114]	3
Indoor air temperature	[165 113 151 74 134]	4
Outdoor air temperature	[151 113 173 65 74]	4

Table 2.30: Top 5 most similar topics for each specific group and their sentiment score

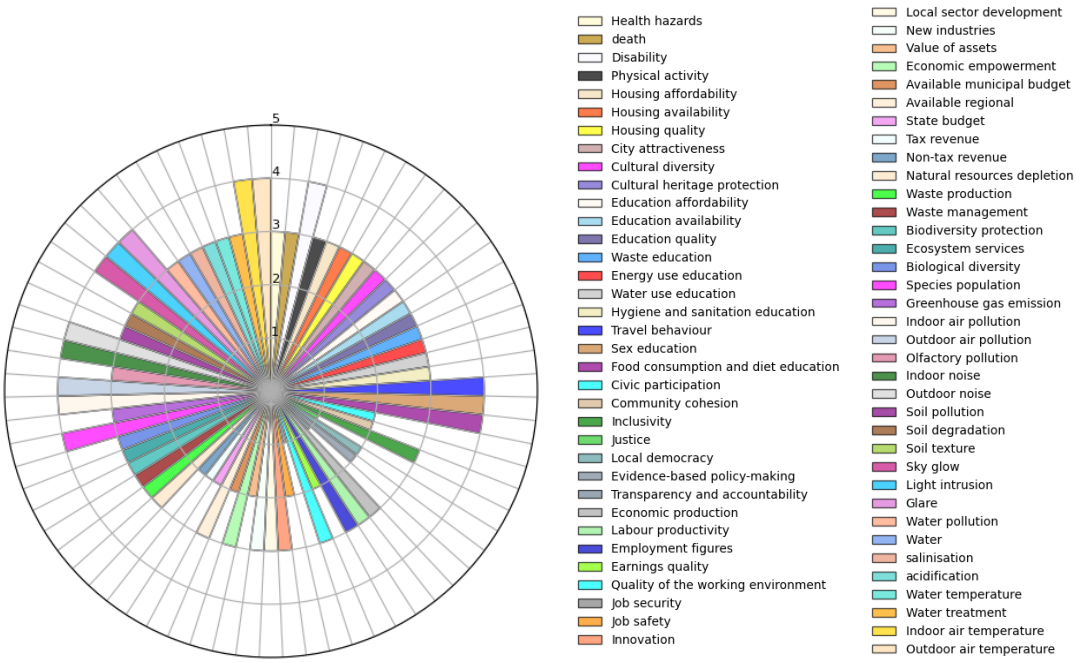


Fig. 2.14: Comparative Sentiment Analysis Scores Across Each Specific Group

2.4.3 Sentiment analysis across Social, Economic, and Environmental Impacts

After acquiring the sentiment analysis results for each distinct group, our next step will be to evaluate the sentiment pertaining to the Social, Economics and Environmental impacts. This process will entail identifying topics closely aligned with these three keywords,

employing a similarity threshold of 0.6 for these keywords and 0.8 for the specific groups associated with each axis, thus ensuring relevance.

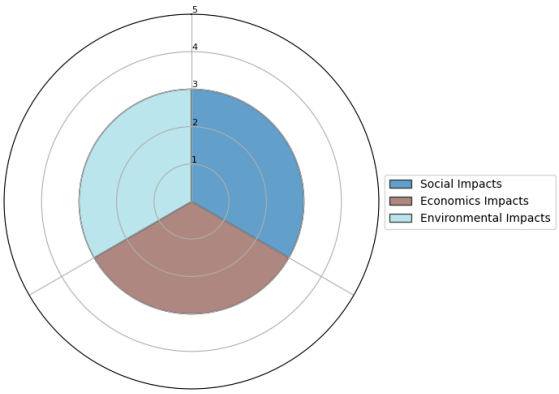


Fig. 2.15: Comparative Sentiment Analysis Scores Across Social, Economic, and Environmental Impact

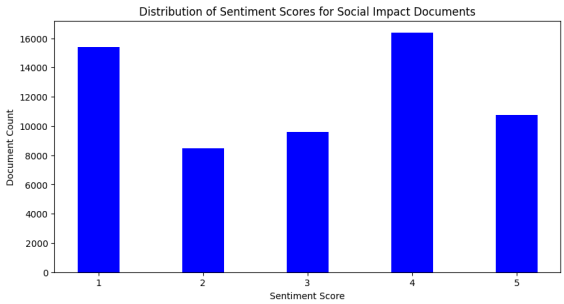


Fig. 2.16: Distribution of Sentiment Scores for Social Impact Documents

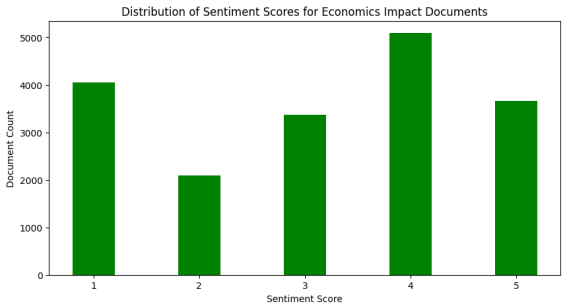


Fig. 2.17: Distribution of Sentiment Scores for Economics Impact Documents

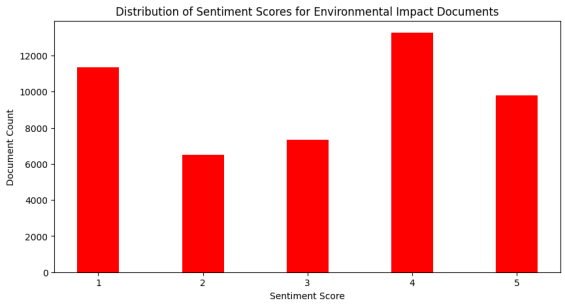


Fig. 2.18: Distribution of Sentiment Scores for Environmental Impact Documents

Based on our three barplots showing the distribution of sentiment scores for social, economic, and environmental impacts, we can derive the following insights:

Social Impacts

- The highest number of documents have a sentiment score of 4 (approximately 16,000), suggesting a predominantly positive sentiment towards social impacts.
- A large number of documents also have a sentiment score of 1 (around 14,000), indicating a substantial amount of negative sentiment.
- Sentiment scores of 3 and 2 have fewer documents, indicating that moderate sentiments are less prevalent.
- The distribution shows a clear bimodal pattern with the highest peaks at very positive (4) and very negative (1) sentiment scores.

Economic Impacts

- The highest number of documents have a sentiment score of 4 (around 5,000), indicating a strong positive sentiment.
- There is a notable number of documents with a sentiment score of 1 (approximately 4,000), indicating a significant portion of negative sentiment.
- The number of documents with a sentiment score of 2 is the lowest, suggesting that slightly negative sentiments are rare in this dataset.
- The distribution shows a similar pattern to the social impacts graph, with peaks at very positive (4) and very negative (1) sentiment scores, but with a generally lower overall document count.

Environmental Impacts

- The highest number of documents have a sentiment score of 4 (approximately 12,500), indicating a generally positive sentiment towards environmental impacts.

- A significant number of documents also have a sentiment score of 1 (around 11,000), showing a notable amount of negative sentiment.
- Sentiment scores of 2 and 3 have fewer documents, suggesting that moderately negative and neutral sentiments are less common compared to extreme sentiments.
- The distribution shows a bimodal pattern with peaks at the very positive (4) and very negative (1) ends of the spectrum.

All three barplots exhibit a bimodal distribution with significant peaks at sentiment scores of 4 and 1, reflecting strong positive and negative sentiments, respectively, across social, economic, and environmental impacts. Moderate sentiments are less prevalent in all three categories.

2.4.4 Sentiment Analysis for SDGs

For the SDGs, we created a new table with the SDGs and extracted the main keywords from each description. Using these keywords, we will identify the 5 topics most similar to each one and then determine the sentiment of the related documents.

objectives	keywords
No Poverty	Poverty
Zero Hunger	Hunger
Good Health and Well-Being	Health, Well-being
Quality Education	Education
Gender Equality	Gender Equality
Quality Education	Water, Sanitation
Affordable and Clean Energy	Affordable Energy, Clean Energy
Decent Work and Economic Growth	Decent Work, Economic growth
Industry, Innovation, and Infrastructure	Industry, Innovation, Infrastructure
Reduced Inequalities	Inequality
Sustainable Cities and Communities	Sustainable cities, Sustainable communities
Responsible Consumption and Production	Responsible consumption, Responsible production
Climate Action	Climate change
Life Below Water	Ocean, Sea, Marine ressources
Life on Land	terrestial ecosystem, Forest, Desertification, land degradation, Biodiversity loss
Peace, Justice and Strong Institutions	Peace, Justice, strong institutions
Partnerships	Global partnership Sustainable Developement

Table 2.31: SDGs objectives and their keywords

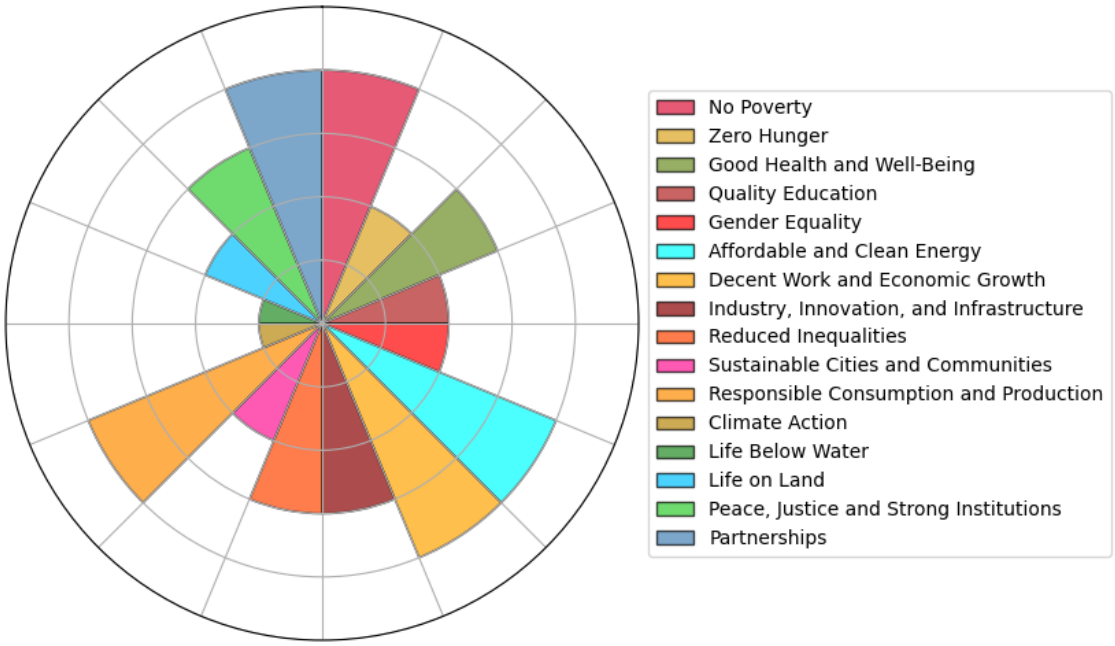


Fig. 2.19: Comparative Sentiment Analysis Scores Across Each SDG Objective

2.5 Limitation of the study

In reflecting on the study's findings and methodologies, several avenues for future research emerge, each offering opportunities to enhance and refine the analytical process.

- Optimization of Model Training Time: The extensive duration required for training later models underscores the need for streamlined hyperparameter tuning processes. For instance, the fifth model demanded a significant 7 hours and 30 minutes for the dimensionality reduction step alone. This prolonged duration poses challenges in experimenting with alternative hyperparameters at a smaller scale, as parameters effective on a reduced scale may not translate optimally to larger datasets, such as determining the minimum cluster size in HDBSCAN or the number of neighbors in UMAP.
- Addressing Memory Constraints: While the use of quantization offers a practical solution to memory constraints, it entails a trade-off in data fidelity, as it represents a form of lossy compression. Investing in improved hardware infrastructure could potentially mitigate the need for quantization, allowing for the utilization of the entire embeddings dataset without compromising on detail.
- Exploring Multilingual Sources: Given that the study exclusively focused on English language sources, future research could extend its scope to encompass diverse linguistic contexts. Incorporating data from social media platforms could offer a broader spectrum of voices beyond traditional press sources, thereby enriching the analysis with diverse perspectives.

- Challenges in Accessing Social Media Data: It is important to acknowledge the evolving landscape of data accessibility, particularly concerning social media platforms like Twitter. Recent changes in scraping policies have rendered the acquisition of real-time data more challenging, highlighting the need for innovative strategies to overcome these obstacles.
- Considerations on Data and Computational Resources: The study underscores the importance of addressing both data and computational limitations. Insufficient data poses a limitation on the robustness of the model, while computational constraints, such as the rate limits imposed by APIs, and the memory requirements of certain algorithms like UMAP, necessitate careful consideration in the design of future studies.
- Sustainability of Model Performance: As embedding models continue to evolve, there arises the challenge of maintaining the relevance and effectiveness of existing models. While newer models may offer enhanced performance, it is important to recognize that this does not render previous models obsolete instantaneously. However, to ensure optimal performance, periodic reevaluation and recalibration of models will be necessary, albeit at the cost of significant time investment.

In summation, these identified areas for future research offer promising avenues for advancing the field, addressing current limitations, and ensuring the continued relevance and efficacy of analytical approaches in the dynamic landscape of natural language processing and data analysis.

Conclusion

This study has investigated the discourse surrounding Environmental, Social, and Governance (ESG) factors and Sustainable Development Goals (SDGs) within African press data using advanced topic modeling and sentiment analysis techniques. The application of BERTopic, a transformer-based topic modeling algorithm, enabled the identification of key themes and patterns in media narratives. This, coupled with sentiment analysis, provided insights into how perceptions of sustainability issues have evolved over time within the African media landscape.

Our findings highlight the diverse and nuanced nature of ESG and SDG discussions, reflecting a broad spectrum of coverage and varying sentiments. The study revealed that African media frequently addresses critical issues related to sustainability, often with a positive or progressive sentiment, indicating a growing awareness and engagement with these topics.

The results of this research contribute to a deeper understanding of the media portrayal and perception of sustainability-related issues in Africa. This understanding is crucial for informing policy-making and corporate strategies, thereby enhancing the effectiveness of initiatives aimed at achieving sustainable development goals across the continent.

Future research could expand on this study by incorporating more recent data and exploring the impact of specific events or policy changes on media discourse. Additionally, comparative studies between different regions or countries within Africa could provide further insights into the local variations in ESG and SDG discussions.

In conclusion, this thesis underscores the importance of leveraging advanced data analysis techniques to decode complex media narratives. It also emphasizes the pivotal role of the media in shaping public discourse around sustainability, which is essential for driving societal change towards achieving sustainable development goals.

Bibliography

[1] The 2030 agenda for sustainable development in france. https://www.agenda-2030.fr/en, accessed: March 31, 2024

[2] Principal component analysis (pca) 101. https://numxl.com/blogs/principal-component-analysis-pca-101/ (2016)

[3] Esg lexicon. https://www.esglawinstitute.com/2021/09/esg-thought-leadership/ (2021)

[4] Attention layers in transformer. https://pylessons.com/transformer-attention (2023)

[5] Amidi, A., Amidi, S.: Recurrent neural networks cheatsheet. Website (2024), https://stanford.edu/~shervine/teaching/cs-230/cheatsheet-recurrent-neural-networks

[6] Ba, J.L., Kiros, J.R., Hinton, G.E.: Layer normalization. arXiv preprint arXiv:1607.06450 (2016)

[7] Barbaresi, A.: Trafilatura: A Web Scraping Library and Command-Line Tool for Text Discovery and Extraction. In: Proceedings of the Joint Conference of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing: System Demonstrations. pp. 122–131. Association for Computational Linguistics (2021), https://aclanthology.org/2021.acl-demo.15

[8] Berenberg, J.: Understanding the sdgs in sustainable investing. Joh Berenberg, Gossler & Co. KG, n. December pp. 1–27 (2018)

[9] Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805 (2018)

[10] Distante, E.: Bertopic: topic modeling as you have never seen it before. https://medium.com/data-reply-it-datatech/bertopic-topic-modeling-as-you-have-never-seen-it-before-abb48bbab2b2 (2022)

[11] Frenzel, C.: On the validation of umap. https://towardsdatascience.com/on-the-validating-umap-embeddings-2c8907588175 (2021)

[12] Grootendorst, M.: Bertopic: Neural topic modeling with a class-based tf-idf procedure. arXiv preprint arXiv:2203.05794 (2022)

[13] Grootendorst, M.: c-tf-idf. https://maartengr.github.io/BERTopic/getting_started/ctfidf/ctfidf.html (2023)

[14] He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 770–778 (2016)

[15] Hochreiter, S., Schmidhuber, J.: Long short-term memory. Neural computation **9**(8), 1735–1780 (1997)

[16] Leland McInnes, John Healy, S.A.: How hdbscan works. https://hdbscan.readthedocs.io/en/latest/how_hdbscan_works.html (2016)

[17] Malzer, C., Baum, M.: A hybrid approach to hierarchical density-based cluster selection. In: 2020 IEEE international conference on multisensor fusion and integration for intelligent systems (MFI). pp. 223–228. IEEE (2020)

[18] McInnes, L.: Parametric (neural network) embedding. https://umap-learn.readthedocs.io/en/latest/parametric_umap.html (2018)

[19] Moulavi, D., Jaskowiak, P.A., Campello, R.J., Zimek, A., Sander, J.: Density-based clustering validation. In: Proceedings of the 2014 SIAM international conference on data mining. pp. 839–847. SIAM (2014)

[20] Muennighoff, N., Tazi, N., Magne, L., Reimers, N.: Mteb: Massive text embedding benchmark. arXiv preprint arXiv:2210.07316 (2022). https://doi.org/10.48550/ARXIV.2210.07316, https://arxiv.org/abs/2210.07316

[21] Reimers, N., Gurevych, I.: Sentence-bert: Sentence embeddings using siamese bert-networks. arXiv preprint arXiv:1908.10084 (2019)

[22] Singh, S.: Countvectorizer vs tfidfvectorizer. https://medium.com/@shandeep92/countvectorizer-vs-tfidfvectorizer-cf62d0a54fa4

[23] Sparck Jones, K.: A statistical interpretation of term specificity and its application in retrieval. Journal of documentation **28**(1), 11–21 (1972)

[24] Supe, K.: Understanding cosine similarity in python with scikit-learn. https://memgraph.com/blog/cosine-similarity-python-scikit-learn (2023)

[25] Ta, A.: Health ai: Is it a pandora's box? https://www.scs.org.sg/bok/ai-ethics?document=101, accessed: March 31, 2024

[26] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L., Polosukhin, I.: Attention is all you need. CoRR **abs/1706.03762** (2017), http://arxiv.org/abs/1706.03762

[27] Venna, J., Kaski, S.: Neighborhood preservation in nonlinear projection methods: An experimental study. In: International conference on artificial neural networks. pp. 485–491. Springer (2001)

[28] Wikipedia contributors: List of african countries by population — Wikipedia, the free encyclopedia (2024), https://en.wikipedia.org/w/index.php?title=List_of_African_countries_by_population&oldid=1214667846, [Online; accessed 31-March-2024]

[29] Wikipedia contributors: Transformer (deep learning architecture) — Wikipedia, the free encyclopedia (2024), https://en.wikipedia.org/w/index.php?title=Transformer_(deep_learning_architecture)&oldid=1217133407, [Online; accessed 6-April-2024]