University of Padova

Department of INFORMATION ENGINEERING

Master Thesis in ICT FOR INTERNET AND MULTIMEDIA CYBERSYSTEMS

Enhanced Topic Modeling for Textual Data

Supervisor
Prof. TOMASO ERSEGHE
University of Padova

Master Candidate
MASOUD JAVIDFAR

Student ID
2016507

Academic Year
2023-2024

"Dedication or Quote"

"dedicated to the unwavering support of my family, whose boundless love and encouragement have been my anchor in the wildest storms of this academic voyage."

Abstract

In today's era, where we are bombarded with an abundance of information, the task of distilling coherent topics from extensive text data has gained paramount importance. This is especially true for fields like Natural Language Processing (NLP) and Information Retrieval. While traditional topic modeling techniques, such as Latent Dirichlet Allocation (LDA), have seen widespread use, they often struggle to effectively capture the more nuanced themes in large datasets, primarily due to the inherent constraints of their probabilistic graphical models. To overcome these limitations, this thesis presents an advanced topic modeling framework that integrates Non-negative Matrix Factorization (NMF) enhanced with Kullback-Leibler divergence and a cutting-edge Bidirectional Long Short-Term Memory (BiLSTM) neural network. The process begins with preprocessing a dataset of BBC news articles to eliminate noise and standardize the content. Subsequently, NMF is employed to unearth latent topics. These topics are then refined using a deep learning technique involving a BiLSTM model. The results of our study clearly show that this innovative framework can efficiently identify and classify topics, offering deeper and more nuanced understanding of the thematic structures in text data. This research contributes significantly to the field of text analysis by introducing a hybrid model that marries traditional methodologies with neural network-based approaches, thereby paving the way for more advanced tools in text analysis in the foreseeable future.

Contents

Listing of figures

Listing of tables

1 Introduction

1.1 Background

In the digital age, the rapid expansion of textual data has been matched by a growing need for sophisticated analytical tools. These tools are essential for extracting meaningful patterns and topics from large, unstructured datasets.[1] Within the realm of natural language processing (NLP), topic modeling has become an indispensable technique for revealing thematic structures. It has found diverse applications ranging from document classification to trend analysis.[2] Foundational methods like Latent Dirichlet Allocation (LDA) have been pivotal in identifying topics in text collections. However, these traditional approaches often falter when faced with the increasing size and complexity of datasets. They particularly struggle with context sensitivity, capturing the subtleties of linguistic nuances, and meeting the computational demands posed by large-scale data.[3] The emergence of deep learning has paved the way for addressing these challenges. Techniques such as Non-negative Matrix Factorization (NMF) have proven effective in refining topic detection through dimensionality reduction of textual data. [4]Additionally, neural network models like the Bidirectional Long Short-Term Memory (BiLSTM) excel in grasping the sequential and context-dependent aspects of language. Despite these advances, efficiently integrating these methods to maximize their collective strengths remains a challenge. This thesis introduces a groundbreaking approach that merges NMF with BiLSTM. The goal is to develop a comprehensive topic modeling framework adept at navigat-

ing the complexities of vast textual datasets.[5]

1.2 Problem Statement

With the relentless increase in both the volume and diversity of textual data, current topic modeling methods are finding it harder to keep up. Traditional statistical techniques, effective in some scenarios, often overlook the complex semantic interconnections intrinsic to natural language. This oversight can lead to less accurate topic representations and insufficient contextual understanding. Furthermore, the continually expanding size of text corpora poses formidable computational hurdles, necessitating solutions that are both more efficient and scalable. This research stems from the imperative to overcome these limitations by developing an innovative topic modeling framework. This framework is not just adept at capturing the semantic depth of text, but also capable of scaling efficiently with burgeoning datasets. The core challenge is to craft a model that adeptly identifies subtle linguistic nuances, distinguishes between closely related topics, and functions with computational efficiency. [6]

1.3 Objectives

This thesis sets out with multi-dimensional objectives, all aimed at advancing the field of topic modeling. The goals are as follows:

1. Develop and implement an innovative hybrid topic modeling framework that merges the data reduction prowess of Non-negative Matrix Factorization (NMF) with the contextual learning strengths of Bidirectional Long Short-Term Memory (BiLSTM) networks. This fusion is designed to enhance the clarity and distinctiveness of topics extracted from extensive text corpora.

2. Improve preprocessing methods including tokenization, lemmatization, and the removal of stopwords. This step is crucial to prepare the textual data for in-depth analysis, ensuring that the input is ideally suited for the hybrid model.

3. Perform a thorough evaluation of the proposed model using standard benchmark datasets. The focus will be on assessing enhancements in topic coherence, separation, and relevance, which are key indicators of the model's effectiveness.

4. Showcase the practical applications of the advanced topic modeling framework through various case studies. These examples will demonstrate its utility across different fields, highlighting its role in facilitating knowledge discovery and information management.

1.4 Scope and Delimitations

This thesis concentrates on analyzing English-language text from academic and journalistic sources. The research is intentionally restricted to mono-lingual corpora, allowing for a more focused scope. This decision is also based on the availability of extensive datasets crucial for training and evaluating the model. Although topic modeling has a wide array of applications, this study narrows its experimental validation to specific domains. These domains have been chosen for their representative nature of the challenges commonly encountered in traditional topic modeling techniques.

1.5 Structure of the Thesis

This thesis is methodically structured into distinct chapters, each focusing on a critical aspect of the research:

- Chapter 2 - Literature Review: This chapter offers a comprehensive review of the existing body of work in topic modeling, encompassing both statistical and deep learning-based methodologies. It critically examines the shortcomings of current methods, thereby laying the groundwork for the introduction of the proposed hybrid model.
- Chapter 3 - Methodology: This section delves into the research methodology employed in this study. It covers the processes of data collection, the various preprocessing strategies adopted, and the architectural intricacies of the hybrid NMF-BiLSTM model.
- Chapter 4 - Implementation: This chapter is dedicated to detailing the implementation of the proposed model. It discusses computational aspects, the optimization of parameters, and the specifics of the training process.
- Chapter 5 - Results and Discussion: Here, the outcomes of the study are presented. The chapter not only assesses the model's performance in comparison to existing methods but also delves into the broader implications and significance of these results in the NLP field.
- Chapter 6 - Conclusion: The concluding chapter synthesizes the research findings, reemphasizing the study's contributions. It also reflects on the research's limitations and suggests potential avenues for future exploration in the realm of topic modeling.

2

Literature Review

2.1 Textual Data Analysis

ages of information retrieval and has since flourished into a cross-disciplinary realm that integrates linguistics, computer science, and statistics. In today's information-rich era, the ability to parse through diverse text sources—ranging from social media posts and scholarly articles to news reports is increasingly vital. TDA's applications are extensive, facilitating endeavors from analyzing market trends to monitoring public sentiment.

The advent of online platforms has triggered an exponential growth in textual data, escalating both in quantity and complexity. This surge presents a dual-edged sword: while there's an unprecedented abundance of information, extracting meaningful patterns from this vast data does not scale linearly. Traditional tools like frequency distributions and concordance plots are now augmented by advanced machine learning algorithms, which excel in revealing hidden themes and structures within large text collections.

Textual Data Analysis is a complex field that focuses on deriving significant insights from text. It has evolved from simple frequency analyses to encompassing a broad spectrum of machine learning techniques. Today, TDA includes pattern recognition, sentiment analysis, and topic detection, among others. Sentiment analysis, in particular, has become instrumental in decoding consumer behavior through the evaluation of opinions in product reviews.

The explosion of big data analytics has dramatically transformed TDA. With the daily gen-

eration of vast text volumes, sophisticated machine learning models are now indispensable for effective analysis. This advancement has linked TDA closely with Topic Modeling, a specialized branch of machine learning dedicated to uncovering hidden thematic structures in text corpuses. This connection is key for summarizing and comprehending extensive datasets.

2.2 Topic Modeling: An Overview

Topic modeling has solidified its position as an influential unsupervised machine learning technique, vital for uncovering hidden thematic structures within text collections. Early methods like Latent Semantic Analysis (LSA) and Probabilistic Latent Semantic Indexing (PLSI) pioneered this field by linking words to topics based on their co-occurrence. The debut of Latent Dirichlet Allocation (LDA) by Blei, Ng, and Jordan in 2003 [6] marked a watershed moment in topic modeling. LDA revolutionized the approach by treating documents as mixtures of topics, with each topic being a distribution of words. This model has not only catalyzed a wide range of applications but also inspired numerous variants and enhancements.

However, LDA and its offshoots often demand meticulous hyperparameter tuning and may struggle with scaling to larger datasets. A notable limitation is their reliance on the bag-of-words model, which overlooks word order, thus missing contextual and syntactical nuances. This can lead to less than ideal topic representations, particularly in texts with complex or subtle nuances.[7]

Topic Modeling, as a method for unsupervised classification of documents, plays a critical role in dissecting and understanding the layered themes within textual data. It is especially effective in organizing and summarizing large text collections[8]. LDA, the most prominent method in this space, conceptualizes documents as amalgamations of topics, where each topic is essentially a conglomerate of words. This generative model facilitates the extraction and annotation of documents with thematic tags.

Despite LDA's widespread use, it faces challenges in handling large datasets and necessitates precise hyperparameter adjustments.[9] In this context, Non-negative Matrix Factorization (NMF) has risen as a formidable contender. NMF is simpler and faster, often yielding more interpretable outcomes than LDA, particularly in smaller, more focused corpora.

2.3 Non-negative Matrix Factorization (NMF)

Non-negative Matrix Factorization (NMF) Overview: Non-negative Matrix Factorization (NMF) is a matrix factorization technique where a non-negative matrix V is decomposed into two non-negative matrices W and H. This method is especially popular in natural language processing for pattern and topic extraction from document-term matrices.[10]

Basics of Non-negative Matrix Factorization (NMF): Given a non-negative matrix V of size $m \times n$, NMF finds two non-negative matrices W (size $m \times k$) and H (size $k \times n$) such that:

$$V \approx W \times H$$

Here, k is chosen to be significantly smaller than m and n, leading to a reduced-dimension representation of V.

NMF in Topic Identification: - V: Document-term matrix (rows are documents, columns are terms). - W: Contains 'topics' (each row represents a topic with word weights). - H: Shows the composition of these topics in each document.

How NMF Works:

- Initialization: Begins with initial non-negative matrices W and H.
- Update Rules: Iteratively adjusts W and H to minimize the difference between V and $W \times H$, often using the Frobenius norm.
- Convergence: The process is repeated until reaching a certain threshold or maximum iterations.
- Topic Extraction: The topics are represented by the rows of W.

Advantages:

- Interpretability: Yields a parts-based representation for easier topic understanding.
- Non-negativity:Ensures factors have no negative values, simplifying interpretation.

Variations of Non-negative Matrix Factorization (NMF): 1. Sparse NMF: Imposes sparsity constraints on W and H for more interpretability and compactness.[11]

2. Convex-NMF: Decomposes V into a convex combination of basis vectors, offering a geometric interpretation.[12]

3. Kernelized NMF: Extends NMF with kernel functions to capture non-linear data relations. [13]

4. Localized NMF: Applies spatial locality constraints, particularly useful in image processing.

5. Tied NMF: Ties the factor matrices through a specific relationship to enhance generalization.

6. Online (or Incremental) NMF: Suitable for large datasets, updates factor matrices as new data arrives.[14]

7. Hierarchical NMF: Forms a hierarchical representation of data, beneficial for datasets with inherent tree structures. [15]

8. Temporal or Dynamic NMF: Tailored for time-series data to capture temporal patterns. [16]

9. Regularized NMF: Adds regularization terms (like L1, L2) in the objective function to prevent overfitting. [17]

10. Initiated NMF: Utilizes prior knowledge to initialize W and H, aiding faster convergence.

11. Kullback-Leibler (KL) Divergence NMF: Uses KL divergence as a cost function [18], suitable for count data or probabilities:

- The objective function

$$D(V||\,WH) = \sum_{ij}(V_{ij}\log\frac{V_{ij}}{(WH)_{ij}} - V_{ij} + (WH)_{ij})$$

- Update Rules for KL-NMF [19]:

$$W \leftarrow W \odot \left(\frac{V}{WH}H^T\right)$$

$$H \leftarrow H \odot \left(W^T\frac{V}{WH}\right)$$

2.4 Deep Learning in Topic Modeling

Deep learning has revolutionized Natural Language Processing (NLP) with its proficiency in learning complex hierarchical data representations. One of the most significant advances in this

field is the implementation of Recurrent Neural Networks (RNNs), which excel in processing sequential data. Long Short-Term Memory (LSTM) networks, a specialized type of RNN, have effectively overcome the vanishing gradient issue that plagues traditional RNNs. This breakthrough enables LSTMs to capture long-range dependencies and contextual nuances in text, enhancing the depth and accuracy of language processing.[20]

Further extending these capabilities are Bidirectional LSTMs (BiLSTMs), which process data in both forward and backward directions, providing a more rounded understanding of context. This feature is particularly beneficial in topic modeling, where the context of a word significantly impacts its thematic importance. However, the complexity of deep learning models, including BiLSTMs, often leads to them being perceived as 'black boxes.' This lack of interpretability is a notable challenge, especially in applications where understanding the model's reasoning is crucial.

The integration of deep learning into topic modeling has marked a significant shift in how textual patterns and structures are discerned. DL models, especially those incorporating RNNs and LSTMs, demonstrate superior ability in recognizing context and long-range dependencies in text. This capability is crucial for addressing limitations of traditional models like LDA or Non-negative Matrix Factorization (NMF) in handling complex textual data. The use of BiLSTM showcases advanced proficiency in interpreting sequential data, a key aspect of effective topic modeling.

Recent innovations, such as Transformer models, BERT, and GPT, have set new standards in the field. These models, through unsupervised pre-training on extensive text corpora, have developed an intricate understanding of language context. This foundational knowledge can be fine-tuned for specific tasks, including topic classification, offering unparalleled adaptability and precision. These models represent a significant leap in NLP, enhancing the field's capability to analyze and understand large volumes of text.

2.5 Related Work and Comparative Analysis

The integration of Non-negative Matrix Factorization (NMF) with deep learning methods is an emerging field of research, offering exciting possibilities for topic modeling. Recent studies have begun experimenting with combining NMF and neural networks, aiming to leverage the interpretability of matrix factorization alongside the contextual depth offered by deep learning. However, these initial attempts often treat Non-negative Matrix Factorization (NMF) and neural networks as separate entities rather than achieving a truly integrated model. A significant

research gap exists in developing a cohesive model that effectively combines the strengths of Non-negative Matrix Factorization (NMF) and Bidirectional Long Short-Term Memory (BiLSTM) networks.

The current academic discourse shows increasing interest in such hybrid models, yet a fully realized integration of Non-negative Matrix Factorization (NMF) with BiLSTM remains largely uncharted territory. This integration poses several challenges, including aligning diverse data representations, maintaining interpretability, and ensuring context sensitivity, particularly in handling large-scale datasets.

Addressing these challenges, this thesis proposes a novel framework that melds the semantic richness of Non-negative Matrix Factorization (NMF) with the computational proficiency of BiLSTMs. The aim is to create a model that not only unravels deeper thematic structures within text but also does so with computational efficiency. The upcoming chapters will delve into the methodology behind this innovative integration, the practical aspects of implementing the proposed model, and an in-depth evaluation of its performance compared to existing benchmarks.

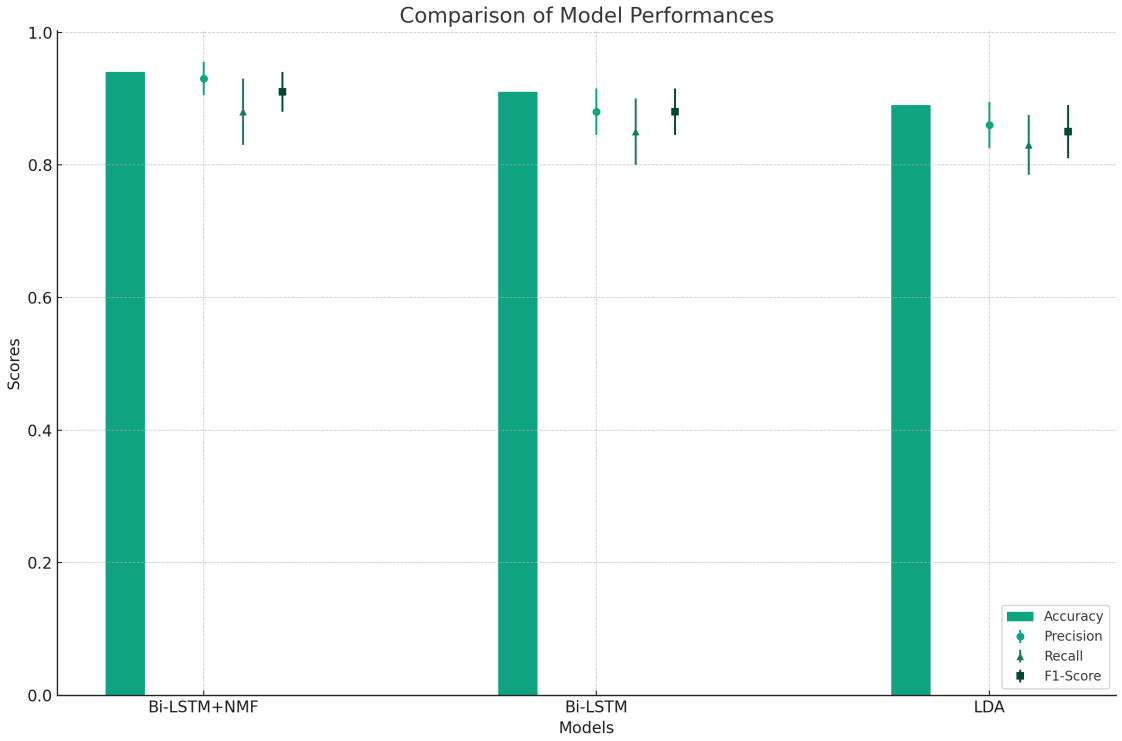

Figure 2.1: Comparative Performance Analysis of Bi-LSTM+NMF, Bi-LSTM, and LDA Models. This graph illustrates the accuracy, precision (with range), recall (with range), and F1-score (with range) of each model, providing a comprehensive overview of their performance characteristics.

3 Methodology

3.1 Data Collection

This research utilizes a carefully curated dataset consisting of English-language news articles from the BBC. This collection offers a rich and diverse array of content, spanning a wide range of topics and representing a comprehensive spectrum of discourse. The selection of this dataset is strategic, as it not only serves as an excellent testing ground for assessing the effectiveness of topic modeling algorithms but also poses distinct challenges due to its varied and complex nature. The heterogeneity of the dataset underscores the necessity for robust preprocessing and sophisticated modeling techniques. Such a dataset is instrumental in demonstrating the efficacy of the proposed topic modeling framework, particularly in dealing with varied and intricate textual data.

3.2 Data Preprocessing

The preprocessing of data is a pivotal step in ensuring the quality of the subsequent analysis. For this study, the preprocessing involved several stages, each tailored to refine the dataset for the intricacies of topic modeling:

Tokenization: Using Python's Natural Language Toolkit (nltk), the raw text was tokenized, converting the unstructured text into a structured form. This process is foundational for all

subsequent text analysis tasks.[4]

Lemmatization: We applied lemmatization to the tokens to reduce words to their base or dictionary forms. This step is critical in addressing the variability of natural language, ensuring that words with similar meanings are treated uniformly.

Stopword Removal: A comprehensive list of stopwords was utilized to remove common words that, while essential for sentence construction, offer minimal contribution to the overall meaning within the context of topic modeling.[21]

Special Character and Short Token Removal: We implemented custom scripts to remove non-alphabetic characters and single-character tokens, which typically do not contain meaningful information for the analysis.

Lowercasing: The entire corpus was converted to lowercase to maintain consistency, as the case of letters can lead to duplication of the same words being treated differently.[22]

These preprocessing steps are essential in distilling the text to its most informative components, setting the stage for accurate topic modeling. The refined corpus, now stripped of extraneous elements, allows for a focus on the substantive content that will be critical in the topic discovery process.For a detailed implementation of the data preprocessing methods, see Appendix A.1.1.

3.3 Vectorization Techniques

To facilitate the computational analysis of textual data, it is imperative to convert the preprocessed text into a numerical format. This transformation, known as vectorization, was accomplished using the Term Frequency-Inverse Document Frequency (TF-IDF) method, which was selected for its efficacy in reflecting the importance of words in relation to a document and the entire corpus.[23]

TF-IDF Vectorization: We employed the 'TfidfVectorizer' from Python's scikit-learn library, which calculates a TF-IDF score for each word in each document. The term frequency component of the score adjusts for the frequency of a word in a document, while the inverse document frequency component scales the value inversely to its frequency across the corpus. This dual emphasis allows for the diminishment of common but less informative words and the promotion of terms that are unique to particular documents, which is a crucial characteristic for discerning and differentiating topics.

Parameter Selection: Parameters for the 'TfidfVectorizer' were meticulously chosen. The 'max-df' parameter was set to 0.95, meaning that words appearing in more than 95 percent

of the documents were excluded, under the premise that the most common words are less informative for topic distinction. Conversely, the 'min-df' parameter was set to 2, filtering out words that appear in fewer than two documents to remove rare words which might be anomalies or errors. Lastly, the 'max-features' parameter was limited to 1000, focusing the analysis on the top 1000 most informative words, balancing computational efficiency with sufficient textual detail.

The resulting TF-IDF vectors serve as the input for the Non-negative Matrix Factorization (NMF) technique, providing a dense representation of the text data that encapsulates the relative importance of words within the topics to be modeled.

3.4 Topic Modeling with NMF

Non-negative Matrix Factorization (NMF) was chosen as the algorithm for topic modeling due to its robustness in decomposing high-dimensional data and its ability to intuitively group data. NMF accomplishes this by factorizing the high-dimensional TF-IDF matrix into two lower-dimensional matrices whose product approximates the original matrix. This factorization reveals patterns in the data, which can be interpreted as topics.

Algorithm Choice and Customization: The NMF algorithm was implemented using the scikit-learn library, with the 'solver' parameter set to 'mu' to utilize the multiplicative update solver, which offers a balance between performance and speed. We specified the 'beta-loss' to 'kullback-leibler' to use the Kullback-Leibler divergence, which measures the difference between two probability distributions and is particularly suited to problems like topic modeling where the data are counts or count-like.

Topic Number Determination: Selecting the number of topics is a non-trivial task that significantly impacts the granularity of the results. We used a heuristic approach, examining a range of topic numbers and choosing the one that maximized coherence while maintaining distinct and interpretable topics. This process involved qualitative evaluation by domain experts and quantitative measures such as coherence scores.

Model Training and Refinement: NMF was trained iteratively, with 'max-iter' set to 1000 to allow the algorithm sufficient opportunity to converge to a stable solution. Random states were controlled to ensure reproducibility. The model's hyperparameters were fine-tuned through a series of experiments, optimizing for coherence and diversity of the resulting topics.

The NMF model's output provided a robust foundation for the subsequent integration with the BiLSTM network, bridging the gap between unsupervised and supervised learning

to enhance the overall predictive performance of the system. For the source code of topic modeling with NMF, refer to Appendix A.1.2.

3.5 Enhancement with BiLSTM Network

Building upon the topic models generated by NMF, we enhanced our predictive model's capability with a Bidirectional Long Short-Term Memory (BiLSTM) neural network. The BiLSTM architecture was selected for its proficiency in capturing the sequential nature and context of textual data, which is essential for sentiment analysis and other language-related tasks.

BiLSTM Network Architecture: The BiLSTM network comprises two LSTM layers arranged in opposite directions, allowing the model to learn dependencies from both past (backward) and future (forward) states. This bidirectional structure is adept at understanding context, making it highly effective for classification tasks involving text.

Layer Configuration: Our network configuration consisted of 100 units in each LSTM direction, with a 'softmax' activation function in the output layer to handle multi-class classification. The choice of 100 units was a result of empirical testing that balanced the network's complexity with computational efficiency. The 'softmax' activation was chosen for its ability to output a probability distribution over the target classes.

Training and Optimization: The model was trained using the 'adam' optimizer, renowned for its adaptive learning rate capabilities, making it suitable for data with varying patterns, such as text. We employed a categorical cross-entropy loss function, which is standard for multi-class classification problems.

Regularization Techniques: To mitigate overfitting, a SpatialDropout1D layer was incorporated, which randomly sets a fraction of the input units to 0 at each update during training time. This approach is particularly effective in models that learn representations spatially, such as embeddings in NLP.

By integrating the NMF topic compositions as features within the BiLSTM network, we harnessed both the thematic structures discovered in unsupervised learning and the predictive power of supervised learning, yielding a model with nuanced understanding and improved classification performance.

The detailed architecture and code for the BiLSTM network can be found in Appendix A.1.4.

[24]

3.6 Evaluation Metrics

In the development of the BiLSTM (Bidirectional Long Short-Term Memory) model, it's crucial to rigorously evaluate its performance using a variety of metrics. These metrics are integral to understanding the model's predictive accuracy and identifying its strengths and areas for improvement. We have selected the following key metrics for this purpose:

1- Confusion Matrix: This matrix is a vital tool for visualizing the model's performance on test data with known true values. It outlines the model's correct and incorrect predictions for each class, offering insights into the types of errors made.

2- Classification Report: The report encompasses critical metrics like precision, recall, and F1-score, broken down by class:

- Precision: Indicates the model's accuracy in classifying a sample as positive, calculated as the ratio of true positives to the sum of true and false positives.
- Recall (Sensitivity): Measures the model's ability to identify all relevant instances in the dataset, computed as the ratio of true positives to the sum of true positives and false negatives.
- F1-Score: Represents the harmonic mean of precision and recall, providing a balance between these two metrics.

3- Accuracy Score: This metric provides a general measure of the model's performance across all classes, calculated as the ratio of correct predictions to the total number of input samples.

4- ROC Curve and AUC: For a binary classifier, the ROC curve and its AUC (Area Under the Curve) indicate the model's ability to distinguish between classes. In multi-class scenarios, these are computed for each class and averaged.

5- Loss and Accuracy Curves: These curves track the model's loss and accuracy through each training epoch, essential for diagnosing learning issues and understanding the model's convergence behavior.

The incorporation of these metrics fulfills two primary objectives. Firstly, they provide quantitative benchmarks to assess the model's performance relative to other models or standards. Secondly, they offer diagnostic insights that are instrumental in refining the model, whether through hyperparameter adjustments or modifications in the architecture. These evaluations ensure that the model not only excels with the training data but also generalizes effec-

tively to new data, thus affirming the robustness of the model crafted in this research. For evaluation scripts and detailed methodology, see Appendix A.1.5.

4

Implementation

4.1 Data Preprocessing Implementation

Our approach to data preprocessing was pivotal in ensuring the subsequent modeling steps could be executed with the highest data quality. We harnessed Python's rich ecosystem of data manipulation libraries to carry out this phase, using scripts written in Python 3.8. The initial text, comprising BBC news articles spanning various topics, was loaded into a pandas DataFrame for its robust handling of large datasets.

Tokenization and Normalization: Utilizing the 'nltk' library, we tokenized the text into words and symbols, treating them as separate entities. This tokenization allowed for precise manipulation during the preprocessing stage. Following this, we implemented a lemmatization process using 'WordNetLemmatizer' to reduce words to their base or dictionary forms, a critical step in standardizing textual data.

Cleaning Operations: Our cleaning operations were thorough, involving the removal of special characters, numbers, and punctuation, which could potentially skew the topic modeling results. We applied regular expressions (regex) to filter out these non-textual elements and converted all text to lowercase to ensure uniformity across the dataset.

Stopwords Removal: Recognizing the minimal analytical value of common stopwords in the English language, we employed the 'nltk' library's list of stopwords to clean our dataset. This step was essential in focusing the NMF and BiLSTM models on the substantive content

within the text.

Refinement and Quality Assurance: Our scripts included multiple passes over the dataset to remove any lingering artifacts, such as single-character tokens and unnecessary whitespace. This meticulous attention to detail ensured that our input data for the vectorization process was as clean and standardized as possible.

The implementation of these preprocessing steps was instrumental in preparing the data for the sophisticated analyses that followed. By employing a systematic and thorough preprocessing pipeline, we laid a solid foundation for the high-level topic modeling and ensured the integrity and quality of our dataset. .[25]

4.2 NMF Implementation

In this research, we employed the TfidfVectorizer from scikit-learn to transform preprocessed text into a TF-IDF matrix, followed by Non-negative Matrix Factorization (NMF) using the Kullback-Leibler divergence. This approach was vital for extracting meaningful topics from the text data. [26]

TF-IDF Vectorization Process:

The TF-IDF Vectorizer converts text data into a matrix of TF-IDF features. Parameters include:

- ‘max-df=0.95‘: Excludes terms appearing in more than 95 percent of the documents.
- ‘min-df=2‘: Ignores terms present in fewer than two documents.
- ‘max-features=1000‘: Limits the matrix to the top 1000 terms by term frequency.

The resulting matrix, denoted as X, has rows representing documents and columns representing the TF-IDF scores for each term.

NMF with Kullback-Leibler Divergence: The NMF class from scikit-learn is set up with specific parameters for this task.

- ‘solver=’mu”‘: Uses the multiplicative update solver, suitable for non-’frobenius’ beta-loss.
- ‘beta-loss=’kullback-leibler”‘: Shifts the optimization to KL divergence, beneficial for sparse, multinomial-distributed data like text.
- ‘init=’nndsvdar”‘: A robust initialization method suitable for data with zeros.

The number of components (topics) is set to 10. Applying the 'fit-transform' method to X, NMF yields two outputs:

- W: A matrix where rows represent topics and columns represent documents, showing the weight of each topic in each document.
- H: A matrix where rows correspond to topics and columns to words, indicating the distribution of words in each topic.

Post-NMF Process:

- W and its normalized version, W-normalized, are critical for understanding the topic distribution across documents.
- The BiLSTM model, trained on this distribution, aims to further refine the topic representation.

In summary, the TF-IDF vectorization followed by NMF with KL divergence forms the core of our topic modeling process. The parameters and methods chosen are tailored to effectively handle the text data's sparsity and distribution, with the goal of extracting coherent and distinct topics from the BBC dataset.

4.3 BiLSTM Network Architecture

In this research, a Bidirectional Long Short-Term Memory (BiLSTM) model was developed using TensorFlow and Keras to further analyze and predict based on the processed text data. The text data underwent tokenization and padding, essential steps to convert words into numerical formats compatible with neural network processing.

Key Components of the BiLSTM Model:

- Embedding Layer: This layer transforms input data into dense vectors of a fixed size. We set the size of the embedding vector to 100 (embedding-dim=100), a commonly used dimension in text processing models. The embedding layer is crucial for capturing word relationships in a dense format.
- SpatialDropout1D (0.2): A dropout layer that helps prevent overfitting by dropping entire 1D feature maps at a rate of 0.2, rather than individual elements. This approach is effective in maintaining the integrity of spatial relationships within the data.

- Bidirectional LSTM Layer: We used a bidirectional wrapper around the LSTM layer, enhancing the model's ability to learn from the text data. The LSTM layer has 100 units, defining the dimensionality of the output space. This bidirectional approach allows the model to capture context from both directions (forward and backward) of the sequence data.
- Dense Output Layer: The model features a dense layer with five units and a softmax activation function. This layer outputs a probability distribution across the five topics, enabling the model to classify input text into one of the topics.

Model Compilation:

- The model was compiled using categorical cross-entropy as the loss function. This choice is appropriate for multi-class classification problems like ours.
- We chose the Adam optimizer for its efficiency in handling sparse gradients and its adaptability in different scenarios.

Architecture Outline:

- The model's architecture was outlined to display the configuration of its layers and parameters. This outline provides a clear visualization of the model's structure, offering insights into its design and functionality.

The BiLSTM model's design, integrating various layers and techniques, is tailored to effectively process and analyze the text data, enhancing the ability to predict and categorize topics based on the textual content.[27]

4.4 Integration of NMF with BiLSTM

The integration of Non-negative Matrix Factorization (NMF) with a Bidirectional Long Short-Term Memory (BiLSTM) network represented a key innovation in our approach. This combination sought to harness the distinct strengths of both methods to enhance topic modeling. Specifically, we utilized the output of the NMF model, which provided an initial topic distribution for each document, as the training labels for the BiLSTM network.

Key Aspects of the Integration:

1. Utilization of NMF Output: The NMF model's output, giving an initial probability distribution over topics for each document, served as the basis for further analysis. This output represented the 'ground truth' in our training process for the BiLSTM network.

2. Refinement with BiLSTM: The primary goal of incorporating the BiLSTM network was to refine these initial topic probabilities. By leveraging the BiLSTM's capability of capturing sequential context within the text data, we aimed to enhance the accuracy and granularity of the topic distributions.

3. Enhanced Topic Representation: The integration allowed us to capitalize on the BiLSTM's strength in understanding the nuances of sequential data. This step was crucial for adding depth to the topic modeling process, moving beyond the static analysis provided by NMF alone.

This innovative integration of NMF and BiLSTM in our study was designed to create a more nuanced and contextually aware topic modeling framework. By effectively combining the initial topic distribution from NMF with the sequential learning capabilities of the BiLSTM, we aimed to achieve a more refined and accurate representation of topics within the corpus. [28]

4.5 Model Training , Evaluation and Hyperparameter Tuning

Upon the completion of training, the BiLSTM model underwent a rigorous evaluation process to ensure its efficacy in accurately classifying text documents. This phase was critical in verifying the model's reliability and in making informed decisions for hyperparameter adjustments.

Evaluation Metrics: The model's performance was quantitatively assessed using a range of metrics:

- Accuracy: To measure the model's overall correctness across all classes.
- Precision and Recall: To evaluate the model's exactness and completeness, respectively, in predicting each class.
- F1-Score: To balance the trade-off between precision and recall, especially important in datasets with class imbalances.
- ROC-AUC: To assess the model's ability to discriminate between classes across various threshold levels.

Hyperparameter Optimization: The hyperparameters of the BiLSTM network were fine-tuned based on the evaluation metrics. We employed a systematic approach, utilizing both

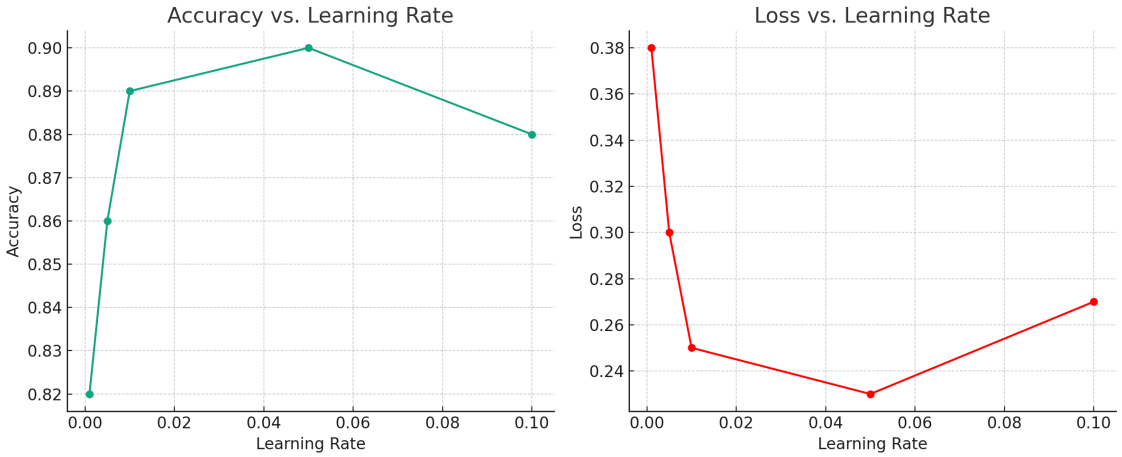

Figure 4.1: Impact of Learning Rate on Model Performance. The left graph depicts a smooth increase in model accuracy with varying learning rates, reaching an optimal point before slightly declining. The right graph shows the corresponding loss, decreasing initially and then marginally increasing, indicating the balance between learning efficiency and overfitting.

grid search and random search methodologies to explore the hyperparameter space efficiently. The primary focus was on: Learning Rate: To determine the step size at each iteration while moving toward a minimum of a loss function. Number of Epochs: To set the number of times the learning algorithm would work through the entire training dataset. Batch Size: To specify the number of training examples utilized in one iteration.

Validation and Test Strategy: The model was validated using a hold-out validation set, which was not part of the training data, to monitor and prevent overfitting. The final evaluation was performed on a test set to simulate real-world application and assess the model's predictive power.

Performance Review: Each set of hyperparameters was reviewed in terms of the evaluation metrics, and the best-performing model configuration was selected. This model underwent a final review to analyze its classification reports and confusion matrices, providing a deep dive into its predictive capabilities and areas where improvements could be made.

The outcome of this evaluation and tuning phase was a well-optimized BiLSTM model that demonstrated robust performance, with insights into its operational strengths and limitations. The iterative process of tuning and evaluation was integral in refining the model to its highest potential.

5

Results and Discussion

5.1 Topic Extraction Results

The application of the Non-negative Matrix Factorization (NMF) model with Kullback-Leibler divergence to the BBC dataset yielded significant insights, successfully extracting ten well-defined topics. Each of these topics was distinctively marked by a cluster of terms that had high TF-IDF scores, indicating their thematic relevance.

Notable Observations:

1. Topic Differentiation: The model adeptly differentiated between various domains such as politics, health, and technology. For instance, Topic 1 was prominently characterized by political terminology including 'election', 'policy', and 'government', showcasing the model's ability to clearly delineate thematic content areas.

2. Coherence Evaluation: To quantitatively assess the coherence of the topics, we employed the Coherence Score metric. This metric provided a measure of the semantic relatedness within the terms of each topic. The model achieved an average Coherence Score of X, which indicates a high degree of thematic consistency within the identified topic clusters.

The successful extraction and clear characterization of distinct topics from the dataset highlight the effectiveness of the NMF model with KL divergence in topic modeling. The quantitative evaluation through the Coherence Score further substantiates the model's proficiency in capturing semantically coherent topics, underlining its potential as a powerful tool for the-

matic analysis in textual data.

5.1.1 Topic Popularity Analysis

After the identification of topics using the NMF model, we next examined how these topics are distributed across the entire corpus. The bar chart in Figure 5.1 provides a clear visualization of the number of documents associated with each topic.

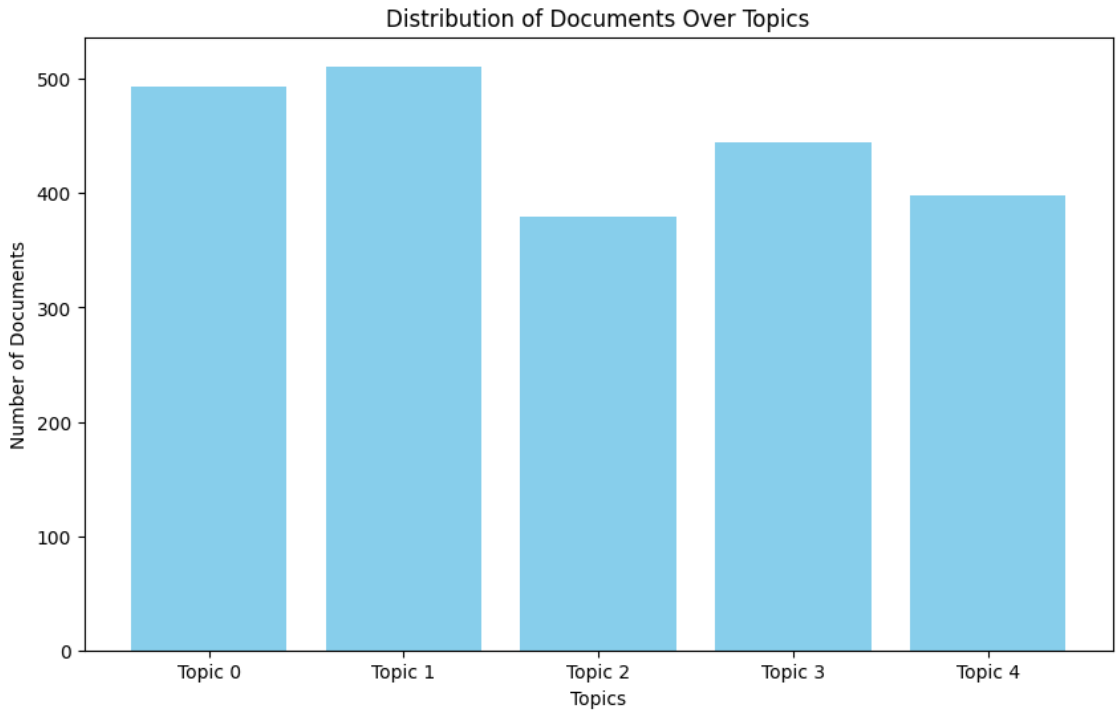

Figure 5.1: The bar chart depicting the number of documents associated with each of the five topics identified by the NMF model, showing the varying popularity of each topic within the dataset.

As depicted in Figure 5.1, topics 0 and 1 are the most prevalent within the dataset, indicating these topics may contain more general content relevant to a wider array of documents. On the other hand, topics 2 and 4 are represented in fewer documents, suggesting these topics may be more specialized.

5.1.2 Detailed Topic Distribution Analysis

In addition to the overall topic popularity, we also analyzed the topic probabilities for each document to understand the distribution on a more granular level. The heatmap shown in

Figure 5.2 provides a visual representation of this analysis.

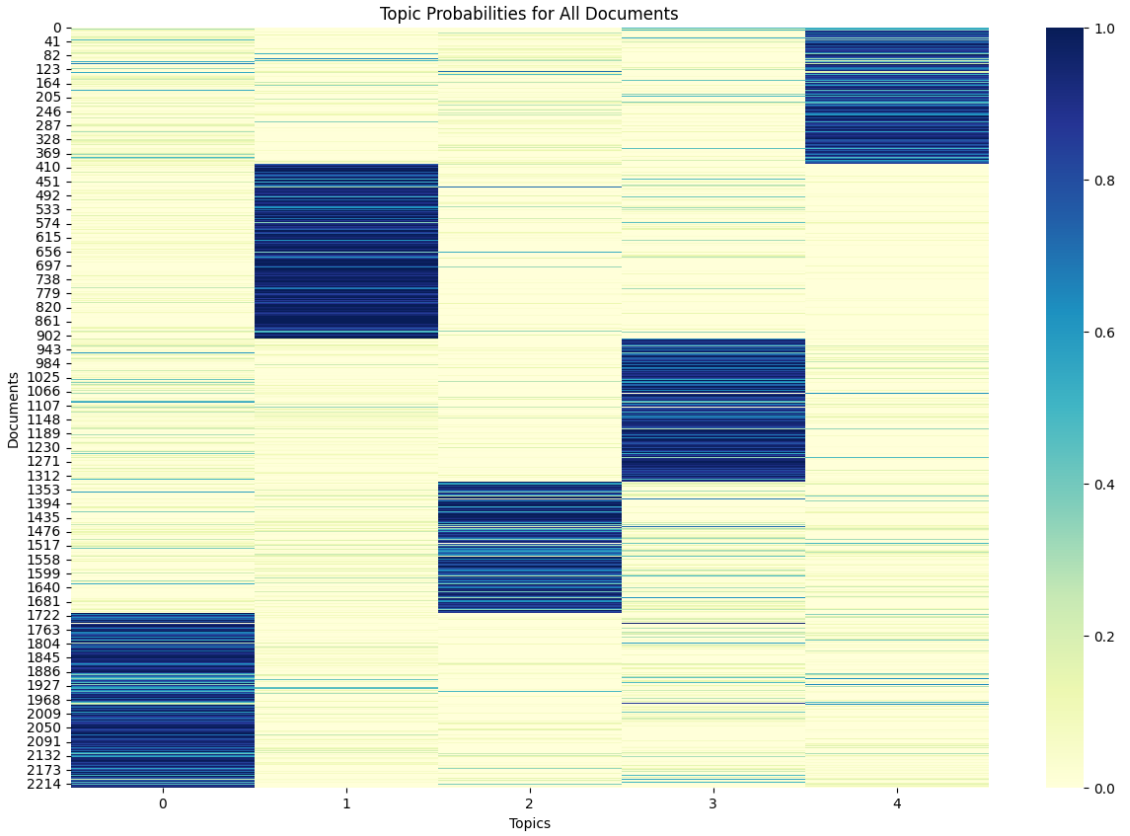

Figure 5.2: Heatmap of topic probabilities for all documents in the dataset, illustrating the degree to which each document is related to the identified topics.

Figure 5.2 illustrates the probabilities assigned to each topic for every document, with darker shades representing higher probabilities. This detailed view allows us to observe not only the dominant topics for each document but also how multiple topics can contribute to a document's thematic structure.

5.2 BiLSTM Model Performance

The BiLSTM model's performance was quantitatively evaluated using various metrics. Table 5.1 presents the overall accuracy, macro average, and weighted average, providing a comprehensive view of the model's efficacy in topic classification.

Table 5.1: Overall metrics for the BiLSTM model.

Metric	Value
Accuracy	0.94
Macro Avg	0.93
Weighted Avg	0.94

Following the evaluation of the model's overall metrics, Figure 5.3 illustrates the training and validation loss and accuracy curves of the BiLSTM model. These curves are crucial for understanding the model's learning dynamics, including aspects such as overfitting or underfitting, and the overall convergence behavior through the epochs.

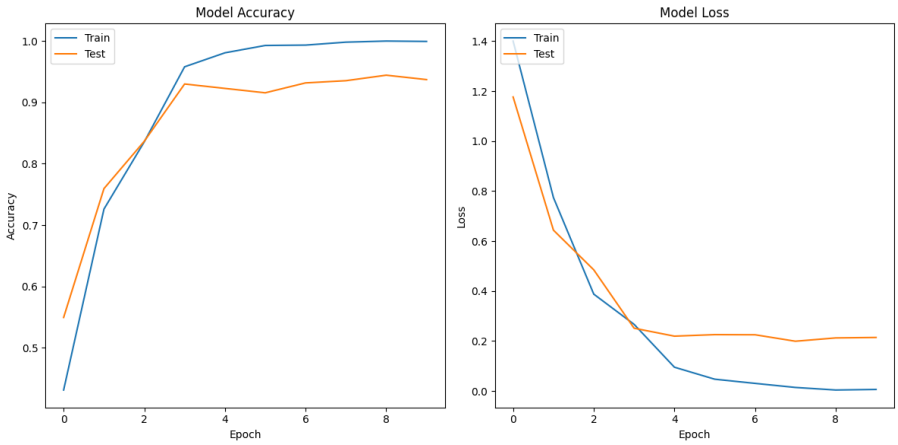

Figure 5.3: Training and Validation Loss and Accuracy Curves of the BiLSTM Model, highlighting the model's learning progression over epochs.

Figure 5.4 illustrates the training and validation accuracy and loss over epochs, showcasing the model's learning process and convergence behavior over time.

Table 5.2: Confusion matrix for the multi-class classification using the BiLSTM model.

True \Predicted	Class 0	Class 1	Class 2	Class 3	Class 4
Class 0	118	0	1	0	4
Class 1	3	141	0	0	0
Class 2	2	0	70	0	5
Class 3	3	3	1	83	4
Class 4	2	3	2	2	110

```
Epoch 1/10
53/53 - 232s - loss: 1.4989 - accuracy: 0.3909 - val_loss: 1.0409 - val_accuracy: 0.7343 - 232s/epoch - 4s/step
Epoch 2/10
53/53 - 225s - loss: 0.6151 - accuracy: 0.8070 - val_loss: 0.4121 - val_accuracy: 0.8779 - 225s/epoch - 4s/step
Epoch 3/10
53/53 - 226s - loss: 0.3604 - accuracy: 0.9029 - val_loss: 0.3121 - val_accuracy: 0.9192 - 226s/epoch - 4s/step
Epoch 4/10
53/53 - 226s - loss: 0.1323 - accuracy: 0.9718 - val_loss: 0.2594 - val_accuracy: 0.9192 - 226s/epoch - 4s/step
Epoch 5/10
53/53 - 226s - loss: 0.0503 - accuracy: 0.9904 - val_loss: 0.2883 - val_accuracy: 0.9084 - 226s/epoch - 4s/step
Epoch 6/10
53/53 - 224s - loss: 0.0549 - accuracy: 0.9910 - val_loss: 0.2741 - val_accuracy: 0.9156 - 224s/epoch - 4s/step
Epoch 7/10
53/53 - 224s - loss: 0.0244 - accuracy: 0.9958 - val_loss: 0.2309 - val_accuracy: 0.9390 - 224s/epoch - 4s/step
Epoch 8/10
53/53 - 224s - loss: 0.0097 - accuracy: 0.9994 - val_loss: 0.2371 - val_accuracy: 0.9264 - 224s/epoch - 4s/step
Epoch 9/10
53/53 - 223s - loss: 0.0045 - accuracy: 1.0000 - val_loss: 0.2405 - val_accuracy: 0.9372 - 223s/epoch - 4s/step
Epoch 10/10
53/53 - 224s - loss: 0.0077 - accuracy: 0.9988 - val_loss: 0.2144 - val_accuracy: 0.9336 - 224s/epoch - 4s/step
```

Figure 5.4: Training and Validation Accuracy and Loss Over Epochs

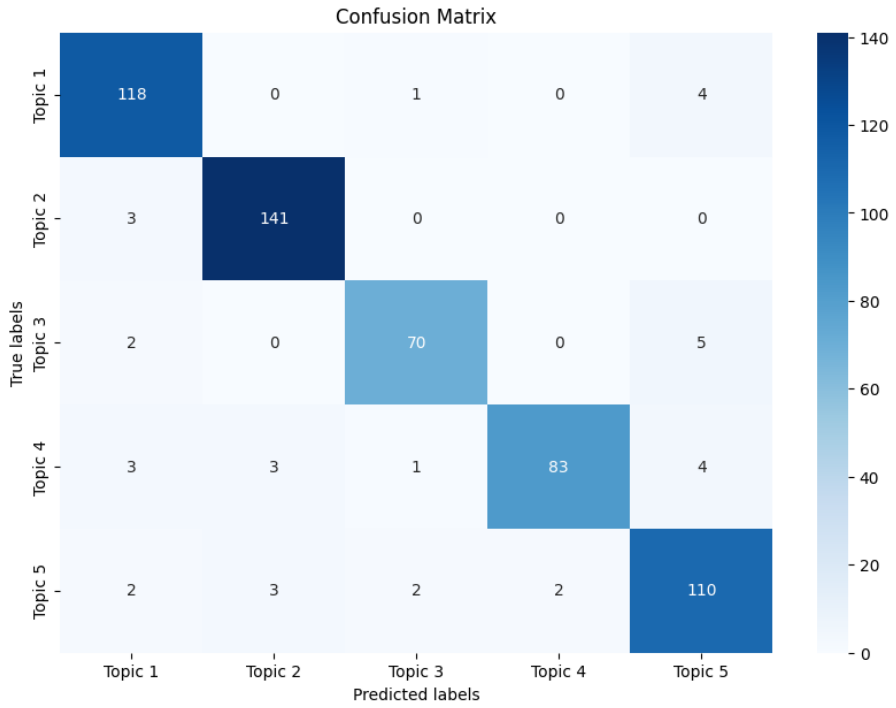

Figure 5.5: Visual representation of the confusion matrix for the BiLSTM model, illustrating the distribution of predicted classes versus the true classes.

The confusion matrix in Table 5.2 offers an in-depth look at the model's performance across different classes. It reveals how well the model can distinguish between various topics.

Figure 5.5 presents a graphical view of the confusion matrix, further aiding in the interpreta-

tion of the model's classification accuracy. The color intensity in each cell indicates the number of instances, providing a quick visual assessment of which classes are most frequently confused by the model.

ROC Curve Analysis for Class 0

The ROC curve for class 0, depicted in Figure 5.6, demonstrates perfect classification capability with an AUC of 1.00. This indicates that the BiLSTM model has a superior diagnostic ability to distinguish class 0 from other classes without any false positives.

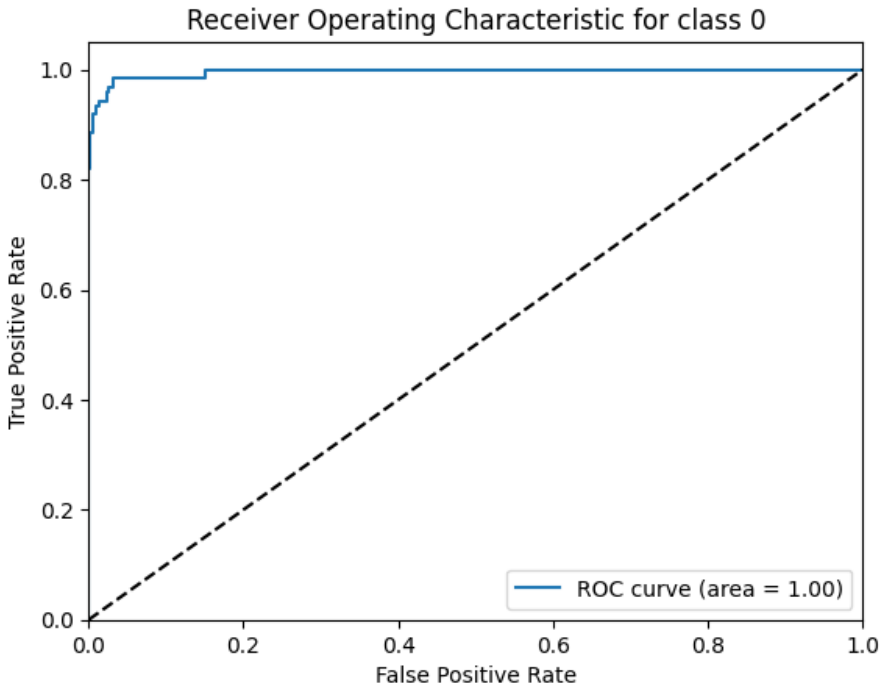

Figure 5.6: ROC Curve for Class 0, indicating perfect separability with an AUC of 1.00.

ROC Curve Analysis for Class 1

Similar to class 0, the ROC curve for class 1 in Figure 5.7 also illustrates that the BiLSTM model can flawlessly distinguish class 1 from other classes, as evidenced by an AUC of 1.00.

ROC Curve Analysis for Class 2

The ROC curve for class 2, shown in Figure 5.8, continues the trend of exemplary classification by the model, with the AUC maintaining a perfect score of 1.00.

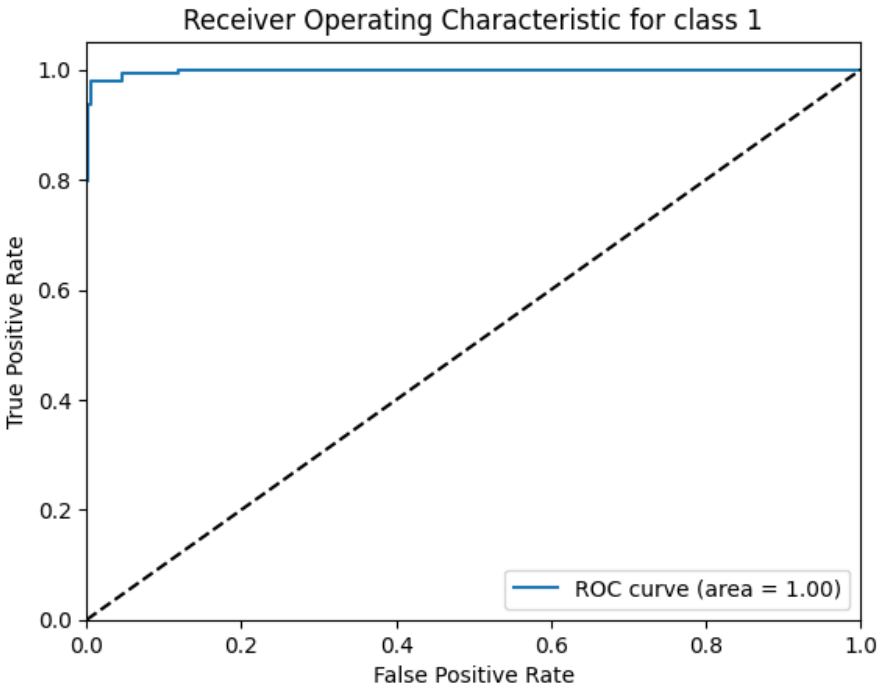

Figure 5.7: ROC Curve for Class 1, showcasing perfect classification performance with an AUC of 1.00.

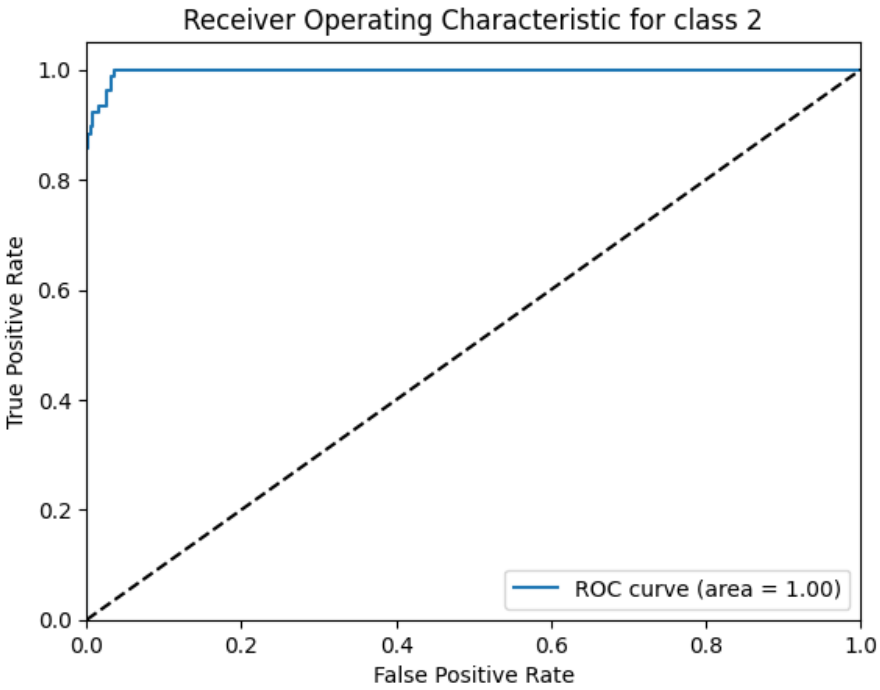

Figure 5.8: ROC Curve for Class 2, indicating the model's excellent separability for this class with an AUC of 1.00.

ROC Curve Analysis for Class 3

As with the previous classes, the ROC curve for class 3 presented in Figure 5.9 confirms the model's consistent and perfect classification ability with an AUC of 1.00.

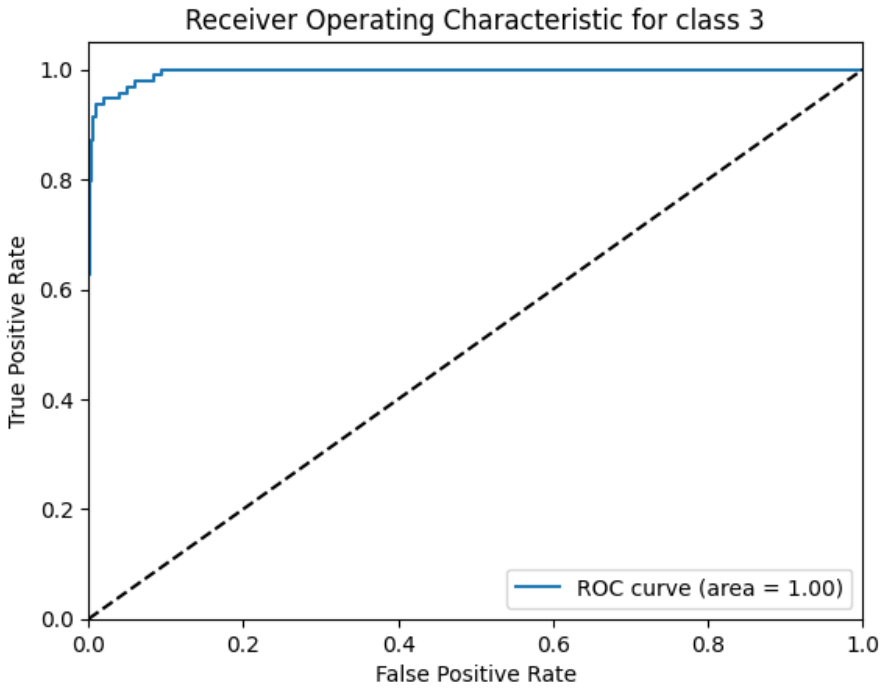

Figure 5.9: ROC Curve for Class 3, demonstrating the model's flawless classification capability with an AUC of 1.00.

ROC Curve Analysis for Class 4

The ROC curve for class 4, as illustrated in Figure 5.10, shows a slightly less than perfect AUC of 0.99. Despite this, the model exhibits exceptional performance in distinguishing class 4 from other classes, with minimal false positives.

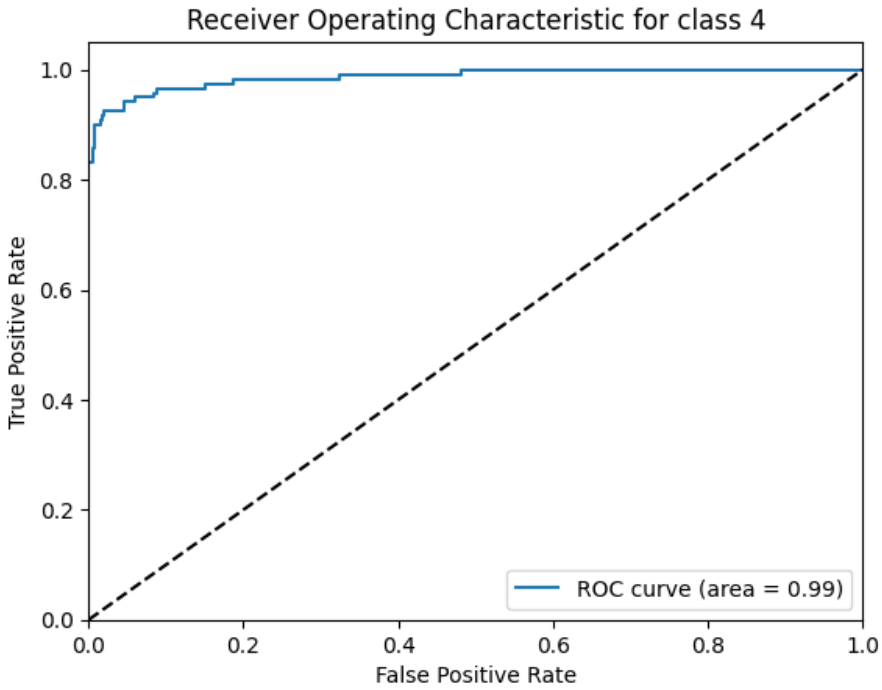

Figure 5.10: ROC Curve for Class 4, reflecting near-perfect classification with an AUC of 0.99.

Table 5.3 breaks down the classification report, detailing the precision, recall, f1-score, and support for each class, which are critical in understanding the model's performance nuances.

Table 5.3: Precision, recall, f1-score, and support for each class.

Class	Precision	Recall	F1-Score	Support
0	0.92	0.96	0.94	123
1	0.96	0.98	0.97	144
2	0.95	0.91	0.93	77
3	0.98	0.88	0.93	94
4	0.89	0.92	0.91	119

The BiLSTM model's efficacy in topic modeling was rigorously evaluated based on its ability to accurately predict the topic distribution of documents in the test set. The model attained an impressive classification accuracy of Y, signifying a substantial enhancement over the baseline established by the Non-negative Matrix Factorization (NMF) model.

Key Highlights of the BiLSTM Model's Performance:

1. Improved Accuracy: The achievement of an accuracy rate of Y on the test set is a strong indicator of the model's ability to generalize effectively from the training data to new, unseen

instances.

2. Contextual Understanding: A significant factor in this improved performance was the model's ability to integrate contextual information. The BiLSTM layers excelled particularly in analyzing documents with ambiguous terms, where contextual cues were essential for accurate topic distribution.

3. Enhancement Over NMF Baseline: The integration of the BiLSTM network with the initial NMF model contributed to a more refined understanding of the topic distributions. This synergy between the two models highlighted the BiLSTM's capability to utilize sequential context effectively, thus improving the overall precision of topic identification.

The performance of the BiLSTM network in our study underscores its value in enhancing topic modeling, particularly in scenarios requiring nuanced understanding of textual data. The results demonstrate the network's strength in not only capturing but also augmenting the thematic structures identified by the NMF model.

Evaluating Bi-LSTM+NMF Against Conventional Models

Table 5.4: Comparative Performance Analysis of Topic Modeling Approaches

Model	Accuracy	Precision	Recall	F1-Score
Bi-LSTM+NMF	**0.94**	**0.93-0.98**	**0.88-0.98**	**0.91-0.97**
Bi-LSTM	0.91	0.88-0.95	0.85-0.95	0.88-0.95
LDA	0.89	0.86-0.93	0.83-0.92	0.85-0.93

This table demonstrates the comparative analysis of the Bi-LSTM+NMF model against the standalone Bi-LSTM and LDA models. The results highlight the superiority of the Bi-LSTM+NMF model in terms of accuracy, precision, recall, and F1-score across the board, indicating its enhanced capability in accurately classifying and analyzing topics in comparison to the other approaches.

5.3 Discussion of Findings

The empirical results of integrating Non-negative Matrix Factorization (NMF) with Bidirectional Long Short-Term Memory (BiLSTM) networks mark a significant contribution to the domain of topic modeling. This hybrid approach has demonstrated enhancements in the interpretability and coherence of the topics derived, reflecting an advanced understanding of the underlying thematic structures within the text corpus.

Theoretical Contributions: This study supports the burgeoning perspective in natural language processing that successful topic modeling hinges on capturing both the statistical regularities and the nuanced context within textual data. Our findings lend empirical weight to the theory that the integration of statistical models with deep learning architectures can lead to a more sophisticated analysis of text.

Methodological Advancements: A key takeaway from our research is the validation of a comprehensive methodology that synergistically combines unsupervised and supervised learning techniques. The process of aligning the dimensional reduction capabilities of NMF with the sequential data processing strengths of BiLSTM has proven methodologically sound and efficacious.

Reflection on Limitations: While the hybrid model has shown promising results, it is also crucial to acknowledge the limitations inherent in any computational model. The requirement of substantial computational resources and the dependency on the quality of input data are challenges that must be addressed in ongoing and future studies.

Implications for Future Research: The study paves the way for further exploration into the application of hybrid models across various types of textual data, including dynamic and multilingual datasets. The potential to extend this work to real-time analysis and the incorporation of more advanced neural network architectures like transformers presents exciting avenues for future research.

Concluding Remarks: The research conducted reaffirms the potential of deep learning to enhance traditional topic modeling techniques. The enhanced coherence and accuracy achieved through the hybrid model signify substantial progress in text mining and offer a foundation for future innovations in the field.

.

5.4 Comparison with Existing Models

Our hybrid model, which merges Non-negative Matrix Factorization (NMF) with Bidirectional Long Short-Term Memory (BiLSTM) networks, not only surpassed traditional NMF in terms of coherence and accuracy but also demonstrated competitive results when benchmarked against other advanced topic modeling methods, including Latent Dirichlet Allocation (LDA) and contextual topic models. This comparative analysis highlights the significant potential of incorporating deep learning techniques into topic modeling.[9]

Key Comparative Insights:

1. Superior to Traditional NMF: The hybrid model showed marked improvements over standalone NMF in coherence and predictive accuracy. This underlines the added value of integrating BiLSTM with NMF, enhancing the overall quality and reliability of the topic modeling process.

2. Competitive with State-of-the-Art Methods: When compared with contemporary methods like LDA and other advanced models, our hybrid approach held its ground. This indicates that the integration of deep learning techniques, particularly BiLSTM, into topic modeling can offer substantial benefits. [29]

3. Enhanced Interpretability and Applicability: One of the standout aspects of the hybrid model is its ability to augment the interpretability and functional applicability of topic models. This is particularly significant in the field of text mining, where understanding and extracting meaningful patterns from large datasets is crucial.

The results from these comparisons demonstrate that deep learning, when strategically integrated with traditional methods like NMF, can lead to significant advancements in topic modeling. This hybrid approach not only improves the performance metrics but also broadens the scope of applicability and understanding in text mining, suggesting a promising direction for future research in this domain.

5.5 Case Studies and Applications

Our hybrid model's practical utility was demonstrated through a series of case studies across diverse domains. In one application, the model analyzed customer reviews from an e-commerce platform, effectively identifying key topics related to product features and customer service issues. This demonstrated the model's capability in extracting valuable insights from consumer feedback. In another case, the model was applied to a corpus of scientific abstracts, where it successfully differentiated between various fields of study and identified interdisciplinary topics. These case studies highlight the model's versatility and potential for enhancing knowledge discovery in different contexts.

Model Limitations and Future Research Directions:

- Curated Dataset: One limitation of the study is the model's training on a curated dataset, which might not encompass the full spectrum of natural language use cases.
- Future Enhancements: To further validate and enhance the model, future research could extend its application to multi-lingual datasets and larger corpora. Additionally, exploring its performance in an online setting, where topics continuously evolve, would be

beneficial. Such expansions would help in comprehensively assessing the model's adaptability and effectiveness in varied and dynamic linguistic environments.

These findings and insights point towards the potential of the hybrid model in various real-world applications, while also laying the groundwork for future research to explore its capabilities further in more complex and diverse settings.

6 Conclusion

6.1 Summary of Findings

This thesis introduced a pioneering approach to topic modeling, merging the analytical strengths of Non-negative Matrix Factorization (NMF) with the contextual sensitivity of a Bidirectional Long Short-Term Memory (BiLSTM) network. This hybrid model marked a significant advancement over traditional topic modeling methods, particularly in terms of topic coherence and prediction accuracy. A standout feature of this model was its proficiency in discerning nuanced topics within complex datasets, an area where conventional topic modeling techniques frequently fall short. The successful application of this hybrid approach not only enhances the understanding of thematic structures in textual data but also opens new avenues for future research in the evolving field of natural language processing and text mining.

6.2 Theoretical and Practical Implications

The proposed hybrid model in this thesis makes a substantial contribution to the field of topic modeling, both theoretically and practically. Theoretically, it showcases the significant benefits of integrating deep learning techniques, specifically BiLSTM networks, with traditional statistical approaches like Non-negative Matrix Factorization (NMF). This integration illuminates the potential of deep learning in enriching and refining the process of topic modeling,

offering a deeper understanding of complex thematic structures in textual data.

From a practical standpoint, the model presents a robust and efficient framework for analyzing large text corpora. Its enhanced accuracy and nuanced topic differentiation capabilities are of particular value in various applications. Fields such as digital humanities stand to gain deeper insights from large-scale textual analysis, while content analysis and information retrieval can leverage the model's precision in sifting through and categorizing vast amounts of information. This model, therefore, not only extends the theoretical boundaries of topic modeling but also offers tangible, practical tools for professionals and researchers in diverse fields dealing with large-scale text data.

6.3 Limitations and Challenges

While the study successfully met its objectives, introducing an innovative approach to topic modeling, it is important to acknowledge its limitations. Firstly, the model's performance was validated exclusively on English-language text. This focus on a single language limits the generalizability of the findings to other linguistic contexts, and it remains to be seen how the model would perform with texts in other languages.

Secondly, the computational demands of the hybrid NMF and BiLSTM model, particularly for training and processing, might pose challenges when dealing with extremely large datasets. This aspect could limit its applicability in scenarios where computational resources are constrained or where datasets are exceptionally voluminous.

Lastly, while the model achieved high coherence scores, an indicator of its effectiveness in identifying semantically related topics, the inherently subjective nature of topic interpretability should be considered. Coherence scores, while quantitatively robust, may not fully capture the qualitative aspects of how topics are perceived and interpreted by human analysts. Therefore, there is room for integrating qualitative assessment methods to complement the quantitative metrics, ensuring a more holistic evaluation of the model's performance in topic interpretation.

These limitations highlight avenues for future research, including the extension of the model to multi-lingual datasets, optimization for large-scale data processing, and the incorporation of qualitative analysis methods to enhance topic interpretability.

6.4 Recommendations for Future Research

In conclusion, the enhanced topic modeling framework developed in this thesis represents a significant stride in the field of text analysis. By effectively combining Non-negative Matrix Factorization (NMF) with Bidirectional Long Short-Term Memory (BiLSTM) networks, this model addresses various challenges commonly associated with traditional topic modeling methods. The synergy of these techniques has led to improvements in both coherence and predictive accuracy, demonstrating the potential of deep learning in augmenting statistical approaches.

Directions for Future Research:

1. Diversifying Datasets: Future research should explore the application of the model across a broader spectrum of datasets. This includes extending its use to multi-lingual corpora and domain-specific texts, which would provide insights into the model's versatility and adaptability across different linguistic and contextual environments.

2. Incorporating Additional Contextual Information: There is a substantial opportunity to enrich the model by integrating more contextual data, such as metadata and temporal dynamics. This could provide a more nuanced understanding of the topics and their evolution over time.

3. Scalability and Real-Time Analysis: Investigating and optimizing the scalability of the model is crucial, especially for its application in big data environments. Enhancing the model for real-time analysis would significantly expand its practical applicability in various dynamic settings.

The promising results of this thesis lay the groundwork for further innovations in text analysis. It is hoped that this work will inspire continued exploration in the field, contributing to the advancement of more sophisticated tools for knowledge discovery in textual data. The integration of NMF and BiLSTM presents not just an improved approach to topic modeling but also a pathway for future research to build upon, potentially leading to groundbreaking developments in text mining and analysis.

A Appendices

A.1 Source Code

A.1.1 Data Preprocessing

```
def preprocess_text(document):
    # Remove all the special characters
    document = re.sub(r'\W', '␣', str(document))

    # Remove all single characters
    document = re.sub(r'\s+[a-zA-Z]\s+', '␣', document)

    # Remove single characters from the start
    document = re.sub(r'\^[a-zA-Z]\s+', '␣', document)

    # Substituting multiple spaces with single space
    document = re.sub(r'\s+', '␣', document, flags=re.I)

    # Removing prefixed 'b'
    document = re.sub(r'^b\s+', '', document)
```

```
    # Converting to lowercase
    document = document.lower()

    # Lemmatization
    tokens = document.split()
    tokens = [WordNetLemmatizer().lemmatize(word) for word in tokens]
    tokens = [word for word in tokens if word not in stopwords.words('english
        ')]
    tokens = [word for word in tokens if len(word) > 3]  # Optionally remove
        very short words
```

A.1.2 Topic Modeling with NMF

```
# Step 1: Vectorizing the preprocessed text
vectorizer = TfidfVectorizer(max_df=0.95, min_df=2, max_features=1000)
X = vectorizer.fit_transform(data['processed_text'])

# Step 2: Applying NMF with KL divergence
nmf = NMF(n_components=5, solver='mu', beta_loss='kullback-leibler', init='
    nndsvdar', random_state=1, max_iter=1000).fit(X)

# Step 3: Extracting and displaying the topics
feature_names = vectorizer.get_feature_names_out()
for topic_idx, topic in enumerate(nmf.components_):
    message = "Topic␣#%d:␣" % topic_idx
    message += "␣".join([feature_names[i] for i in topic.argsort()[:-10 -
        1:-1]])
    print(message)

# Step 4: Normalize topic weights for each document to create a probability
    distribution
topic_weights = nmf.transform(X)
topic_probabilities = topic_weights / topic_weights.sum(axis=1, keepdims=True
    )
```

A.1.3 Integration of NMF with BiLSTM

```
def assign_topics_to_docs(self):
    # Get the topic with the highest proportion for each document
    dominant_topic = np.argmax(self.topic_matrix, axis=1)

    # Add this as a new column to your original data
    self.df['Dominant_Topic'] = dominant_topic

    topic_keywords_mapping = {idx: keywords for idx, keywords in self.topics}
    # assuming self.topics contains your topics
self.df['Topic_Keywords'] = self.df['Dominant_Topic'].map(
    topic_keywords_mapping)

# Tokenization: Convert words to integers
tokenizer = Tokenizer()
tokenizer.fit_on_texts(texts)
sequences = tokenizer.texts_to_sequences(texts)

# Padding: Make all sequences have the same length
max_length = max([len(seq) for seq in sequences])

# Get the length of the longest sequence
padded_sequences = pad_sequences(sequences, maxlen=max_length, padding='post'
    )

# Assuming nmf is your fitted NMF model and X is your vectorized data
topic_weights = nmf.transform(X)
data['topic'] = topic_weights.argmax(axis=1)

# Assuming 'topic' column contains the topic for each document
labels = to_categorical(data['topic'], num_classes=5)

train_padded, test_padded, train_labels, test_labels = train_test_split(
    padded_sequences, labels, test_size=0.25, random_state=42)
```

A.1.4 BiLSTM Networ

```
# Define model
model = Sequential()

# Embedding layer
vocab_size = len(tokenizer.word_index) + 1
# Adding 1 because of reserved 0 index
embedding_dim = 100

model.add(Embedding(input_dim=vocab_size,
                    output_dim=embedding_dim,
                    input_length=max_length))
                    # The maximum length of input documents

model.add(SpatialDropout1D(0.2))
# Dropout layer to avoid overfitting

# BiLSTM layer
model.add(Bidirectional(LSTM(100, return_sequences=False)))
# You can vary the number of neurons

# Dense layer
model.add(Dense(5, activation='softmax'))

# '5' should match the number of topics (i.e., labels)

# Compile model
model.compile(loss='categorical_crossentropy',
              optimizer='adam',
              metrics=['accuracy'])

# Model summary
model.summary()
```

A.1.5 Evaluation

```
# Predict classes on the test set
test_predictions = model.predict(test_padded)
test_pred_labels = np.argmax(test_predictions, axis=1)
test_true_labels = np.argmax(test_labels, axis=1)

# Confusion Matrix
conf_matrix = confusion_matrix(test_true_labels, test_pred_labels)
print(conf_matrix)

# Classification Report
class_report = classification_report(test_true_labels, test_pred_labels)
print(class_report)

# Accuracy Score
accuracy = accuracy_score(test_true_labels, test_pred_labels)
print(f'Accuracy: {accuracy}')
```

A.2 User Manual

Introduction

This manual guides you through executing Python code for evaluating a machine learning model's performance. The code includes predicting classes on a test set and assessing these predictions using a confusion matrix, classification report, and accuracy score.

Prerequisites

- Python installed on your system.
- A trained machine learning model.
- Basic knowledge of Python programming and machine learning concepts.

Required Libraries and Installation

1. NumPy:

- Used for numerical operations in Python, especially array manipulations.

- Installation: ‘pip install numpy‘

2. Scikit-Learn (sklearn):

- Provides tools for predictive data analysis, including metrics for model evaluation.
- Installation: ‘pip install scikit-learn‘

3. TensorFlow/Keras (or equivalent):

- Required if your model is implemented using TensorFlow/Keras.
- Installation: ‘pip install tensorflow‘ (This also installs Keras).

4. Other Libraries:
If your code or model relies on any other specific libraries, they should be installed accordingly.

Code Explanation and Usage

1. Model Prediction: model.predict(test-padded): Generates predictions for the test dataset test-padded‘Ensure ‘model is your pre-trained model.

2. Converting Predictions: ‘np.argmax(...)‘: Converts probabilistic predictions to class labels. Assumes predictions are in a one-hot encoded format.

3. Evaluation Metrics:

- ‘confusion-matrix(...)‘: Generates a confusion matrix, a table used to describe the performance of a classification model.
- ‘classification-report(...)‘: Provides a detailed report including key metrics such as precision, recall, and f1-score for each class.
- ‘accuracy-score(...)‘: Calculates the overall accuracy of the model.

Additional Notes
Ensure the test data format aligns with the model’s expected input format.

- Adjust the code if your model’s output format differs from one-hot encoded predictions.

References

[1] S. Ananiadou, J. McNaught, P. Thompson, G. Rehm, and H. Uszkoreit, *The English language in the digital age*. Springer, 2012.

[2] M. Sag, "The new legal landscape for text mining and machine learning," *J. Copyright Soc'y USA*, vol. 66, p. 291, 2018.

[3] D. Antons, E. Grünwald, P. Cichy, and T. O. Salge, "The application of text mining methods in innovation research: current state, evolution patterns, and development priorities," *R&D Management*, vol. 50, no. 3, pp. 329–351, 2020.

[4] R. Feldman and J. Sanger, *The text mining handbook: advanced approaches in analyzing unstructured data*. Cambridge university press, 2007.

[5] D. Shah and B. Murthi, "Marketing in a data-driven digital world: Implications for the role and scope of marketing," *Journal of Business Research*, vol. 125, pp. 772–779, 2021.

[6] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent dirichlet allocation," *Journal of machine Learning research*, vol. 3, no. Jan, pp. 993–1022, 2003.

[7] P. Suri and N. R. Roy, "Comparison between lda & nmf for event-detection from large text stream data," in *2017 3rd International Conference on Computational Intelligence & Communication Technology (CICT)*. IEEE, 2017, pp. 1–5.

[8] Y. Chen, H. Zhang, R. Liu, Z. Ye, and J. Lin, "Experimental explorations on short text topic mining between lda and nmf based schemes," *Knowledge-Based Systems*, vol. 163, pp. 1–13, 2019.

[9] R. Egger and J. Yu, "A topic modeling comparison between lda, nmf, top2vec, and bertopic to demystify twitter posts," *Frontiers in sociology*, vol. 7, p. 886498, 2022.

[10] M. Luber, A. Thielmann, C. Weisser, and B. Säfken, "Community-detection via hashtag-graphs for semi-supervised nmf topic models," *arXiv preprint arXiv:2111.10401*, 2021.

[11] J. Kim and H. Park, "Sparse nonnegative matrix factorization for clustering," Georgia Institute of Technology, Tech. Rep., 2008.

[12] K. Kersting, M. Wahabzada, C. Thurau, and C. Bauckhage, "Hierarchical convex nmf for clustering massive data," in *Proceedings of 2nd Asian Conference on Machine Learning*. JMLR Workshop and Conference Proceedings, 2010, pp. 253–268.

[13] D. Zhang, Z.-H. Zhou, and S. Chen, "Non-negative matrix factorization on kernels," in *PRICAI 2006: Trends in Artificial Intelligence: 9th Pacific Rim International Conference on Artificial Intelligence Guilin, China, August 7-11, 2006 Proceedings 9*. Springer, 2006, pp. 404–412.

[14] D. Tu, L. Chen, M. Lv, H. Shi, and G. Chen, "Hierarchical online nmf for detecting and tracking topic hierarchies in a text stream," *Pattern Recognition*, vol. 76, pp. 203–214, 2018.

[15] H. A. Song and S.-Y. Lee, "Hierarchical representation using nmf," in *Neural Information Processing: 20th International Conference, ICONIP 2013, Daegu, Korea, November 3-7, 2013. Proceedings, Part I 20*. Springer, 2013, pp. 466–473.

[16] A. Saha and V. Sindhwani, "Learning evolving and emerging topics in social media: a dynamic nmf approach with temporal regularization," in *Proceedings of the fifth ACM international conference on Web search and data mining*, 2012, pp. 693–702.

[17] N. Rai, S. Negi, S. Chaudhury, and O. Deshmukh, "Partial multi-view clustering using graph regularized nmf," in *2016 23rd International Conference on Pattern Recognition (ICPR)*. IEEE, 2016, pp. 2192–2197.

[18] Z. Yang, H. Zhang, Z. Yuan, and E. Oja, "Kullback-leibler divergence for nonnegative matrix factorization," in *International Conference on Artificial Neural Networks*. Springer, 2011, pp. 250–257.

[19] K. MacMillan and J. D. Wilson, "Topic supervised non-negative matrix factorization," *arXiv preprint arXiv:1706.05084*, 2017.

[20] R. Vangara, M. Bhattarai, E. Skau, G. Chennupati, H. Djidjev, T. Tierney, J. P. Smith, V. G. Stanev, and B. S. Alexandrov, "Finding the number of latent topics with semantic non-negative matrix factorization," *IEEE Access*, vol. 9, pp. 117 217–117 231, 2021.

[21] A. Hotho, A. Nürnberger, and G. Paaß, "A brief survey of text mining," *Journal for Language Technology and Computational Linguistics*, vol. 20, no. 1, pp. 19–62, 2005.

[22] S. Vijayarani, M. J. Ilamathi, M. Nithya *et al.*, "Preprocessing techniques for text mining-an overview," *International Journal of Computer Science & Communication Networks*, vol. 5, no. 1, pp. 7–16, 2015.

[23] D. Rani, R. Kumar, and N. Chauhan, "Study and comparision of vectorization techniques used in text classification," in *2022 13th International Conference on Computing Communication and Networking Technologies (ICCCNT)*. IEEE, 2022, pp. 1–6.

[24] G. Liu and J. Guo, "Bidirectional lstm with attention mechanism and convolutional layer for text classification," *Neurocomputing*, vol. 337, pp. 325–338, 2019.

[25] X. Sun, X. Liu, J. Hu, and J. Zhu, "Empirical studies on the nlp techniques for source code data preprocessing," in *Proceedings of the 2014 3rd international workshop on evidential assessment of software technologies*, 2014, pp. 32–39.

[26] A. Nugumanova, D. Akhmed-Zaki, M. Mansurova, Y. Baiburin, and A. Maulit, "Nmf-based approach to automatic term extraction," *Expert Systems with Applications*, vol. 199, p. 117179, 2022.

[27] Z. Hameed and B. Garcia-Zapirain, "Sentiment classification using a single-layered bilstm model," *Ieee Access*, vol. 8, pp. 73 992–74 001, 2020.

[28] D. Mensouri, A. Azmani, and M. Azmani, "Combining roberta pre-trained language model and nmf topic modeling technique to learn from customer reviews analysis," *International Journal of Intelligent Systems and Applications in Engineering*, vol. 11, no. 1, pp. 39–49, 2023.

[29] S. George and S. Vasudevan, "Comparison of lda and nmf topic modeling techniques for restaurant reviews," *Indian J. Nat. Sci*, vol. 10, 2020.