Technische Hochschule Deggendorf
Fakultät Angewandte Informatik

Studiengang Master Artificial Intelligence and Data Science

Erforschung der semantischen Homogenität beim Clustering Ungelabelt Daten mithilfe grosser Sprachmodelle

Exploring Semantic Homogeneity in Unlabeled Data Clustering Using Large Language Models

Masterarbeit zur Erlangung des akademischen Grades:

Master of Science (M.Sc.)

an der Technischen Hochschule Deggendorf

Vorgelegt von:

Bashar Fares

Matrikelnummer: 22109805

Am: 31. Jan 2024

Prüfungsleitung:

Prof. Dr. Andreas Fischer

Ergänzende Prüfende:

Zineddine Bettouche

Erklärung

Name des Studierenden: Bashar Fares

Name des Betreuenden: Prof. Dr. Andreas Fischer

Thema der Abschlussarbeit:

Erforschung der semantischen Homogenität beim Clustering Ungelabelt Daten mithilfe großer Sprachmodelle

1. Ich erkläre hiermit, dass ich die Abschlussarbeit gemäß § 35 Abs. 7 RaPO (Rahmenprüfungsordnung für die Fachhochschulen in Bayern, BayRS 2210-4-1-4-1-WFK) selbständig verfasst, noch nicht anderweitig für Prüfungszwecke vorgelegt, keine anderen als die angegebenen Quellen oder Hilfsmittel benutzt sowie wörtliche und sinngemäße Zitate als solche gekennzeichnet habe.
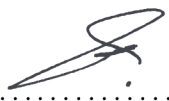
   Deggendorf,   31.01.2024
   ....
            Datum   Unterschrift des Studierenden

2. Ich bin damit einverstanden, dass die von mir angefertigte Abschlussarbeit über die Bibliothek der Hochschule einer breiteren Öffentlichkeit zugänglich gemacht wird:

   ○ Nein

   ⦿ Ja, nach Abschluss des Prüfungsverfahrens

   ○ Ja, nach Ablauf einer Sperrfrist von … Jahren.

   Deggendorf,   31.01.2024
   ....
            Datum   Unterschrift des Studierenden

Bei Einverständnis des Verfassenden vom Betreuenden auszufüllen:

Eine Aufnahme eines Exemplars der Abschlussarbeit in den Bestand der Bibliothek und die Ausleihe des Exemplars wird:

○ Befürwortet

○ Nicht befürwortet

Deggendorf, ...
            Datum   Unterschrift des Betreuenden

Abstract

In recent years, Transformer models have emerged as powerful tools for contextualized text representation, achieving state-of-the-art performances. Particularly in unsupervised settings, these models prove useful when labeled data is unavailable. This thesis investigates the topical clustering of unlabeled scientific text, leveraging various pre-trained large language models. The primary focus is on grouping the publication database at Deggendorf Institute of Technology (DIT) according to their main topics.

The initial experiments employ the BERT-base model, established as the baseline approach. The study also explores the effectiveness of TinyLlama, a compact 1.1B parameters chat model adopting the Llama2 architecture, demonstrating its high potential in revealing diverse topics within the dataset. Additionally, the implications of using encoded Abstract Meaning Representations (AMR) are explored, especially in the context of encoding the publications with an AMR parser. The study further investigates the advantages of reducing the dimensionality of BERT encodings into a lower space through the application of autoencoders.

The experiments showcase the efficiency of TinyLlama and the reduced set of BERT encodings in the task of topical modeling, favoring these methods over the traditional approach using BERT. This research contributes to the growing field of unsupervised topical clustering, offering insights and methodologies for efficient exploration and understanding of scientific text. The code is available here https://mygit.th-deg.de/bf01805/thesis.git

Contents

1 Introduction

In the last few years, pre-trained Large Language Models have revolutionized the field of Natural Language Processing with their ability to produce contextualized vector representations, as well as capture long-range dependencies in the input text. These encodings can be utilized in down-stream NLP tasks, such as translation, text classification, question answering, clustering, all the way to text generation. Different variations of these models have achieved state-of-the-art performances on different tasks, whether they were based on the full transformer architecture (e.g. Llama, BART), encoder-only models (e.g. BERT), or the ones based on the decoder component of the transformer (e.g. GPT).

1.1 Motivation

In the task of topical clustering of unlabeled text, researchers have employed variations of BERT model to acquire embeddings that can be later utilized for clustering. However, different challenges may arise when dealing with complex and mixed scientific domains, since long and complex scientific terminologies affect the efficiency of the produced embeddings and their ability to form meaningful and well-separated clusters.

Another common issue that usually affects the clustering performance is the Curse of Dimensionality effect. Transformer models encode the input into a high-dimensional space, for instance, BERT-base encodes the input text into a 768-dimensional space. This high dimensionality in the produced embeddings can negatively affect the performance of clustering algorithms such as k-means, when too many features lead the algorithm to lose sense of the relative distances between data points, resulting in insufficient clustering.

1.2 Objective

This thesis work explores the possible ways to perform topical clustering of scientific research texts after encoding them using transformer models. The aim is to explore ways to minimize the negative impact of semantically complex text, as well as reduce the negative effect of the high dimensionality by using dimensionality reduction techniques. The experiments include the use of different types of Transformer models, including the basline approach using BERT and mBERT (multilingual BERT), AMR Parser, as well as TinyLlama-1.1B-Chat which is a compact chat model that adopts the architecture of Llama2.

The used models are different in their architecture and learning objectives. Initially, the common way of encoding text using BERT is performed. The BERT encodings are going to serve as a benchmark to compare with encodings coming from the chat model TinyLlama, as well as encodings coming from an Abstract Meaning Representation (AMR) parser.

This study also explores the advantages of employing deep auto-encoders to reduce the dimensional space of encodings for enhanced k-means clustering, aiming to minimize information loss during the reduction process.

The investigation addresses three key questions:

1. Can TinyLlama, a 1.1B parameter chat model adopting the architecture of Llama2, outperform the baseline approach of using Bidirectional Encoders (BERT) in the task of topical clustering of scientific publications?

2. What are the implications of using encoded Abstract Meaning Representations, particularly when encoding text using an AMR parser?

3. Can autoencoders introduce improvements through their ability to encode input data into a lower-dimensional space?

This analysis tackles these key questions with a focus on clustering the scientific publications within the Deggendorf Institute of Technology (DIT) publication database, covering various topics. The objective is to group these documents based on their primary themes. The experiments with TinyLlama enabled the discovery of 14 detailed topics while achieving clustering scores comparable to the common approach using BERT encodings, which enables the clustering of 8 topics. Furthermore, the proposed reduced BERT encodings introduced significant improvements in clustering scores, achieving score improvement of 116%, 135%, and 38% in Silhouette, Calinski-Harabasz, and Davies-Bouldin scores respectively, while allowing the discovery of 13 different topics. Determining the theme of a cluster involves analyzing the most frequent and relevant keywords extracted from the documents within that cluster, where Key-BERT is utilized for that purpose, helping identify and highlight the key topics characterizing the cluster's content.

1.3 Thesis Structure

With regards to the organization of this thesis, the **B**ackground chapter gives a brief description of the main concepts and technologies relevant to this work, including the Transformer models, clustering, dimensionality reduction, Abstract Meaning Representation, and evaluation metrics. **R**elated Work chapter explores relevant previous work. The chapter **M**ethodology starts by performing basic exploratory analysis on the the dataset utilized in this work, and then goes through the methods used in this work in a more detailed approach. The **E**xperiments chapter shows the final results and compares the different models and techniques followed to perform the clustering. The **R**esults Discussion chapter recaps the results and summarize them in a table. Finally, the **C**onclusion chapter summarizes the final take-away from this work.

2 Background

This chapter goes through the main concepts and technologies used in this work, providing a brief description of the Transformers, K-means clustering, Abstract Meaning Representation Graphs, Cluster Evaluation Metrics, Keyword Extraction using KeyBERT, Deep Autoencoders, and dimensionality reduction using UMAP.

2.1 Transformers

The Transformer, introduced in the famous paper "Attention is All You Need" by Vaswani et al. [1], has revolutionized Natural Language Processing. Its encoder-decoder architecture and self-attention mechanism enables parallelization, and capturing long-range dependencies in language modeling tasks. The model proposed in the paper is shown in Figure 2.1. The encoder maps an input sequence to continuous representations in a high-dimensional space, which is then processed by the decoder. The decoder receives the output of the encoder together with the decoder output at the previous time step to generate an output sequence. This architecture opened doors for many different applications, achieving state-of-the-art performances on a wide range of NLP tasks.

The high-dimensional embeddings generated by the encoder component of the transformer are leveraged in this study, since the encoder is capable of capturing the contextualized meaning of the input sentences. In terms of the architecture of the models used in this thesis, encoder-only models (e.g. BERT), as well as encoder-decoder models (e.g. BART, Tiny Llama) are explored. The following subsections discuss the specifics of each used model, its distinctive properties, and the rationale behind its selection.

2.1.1 BERT-base-uncased

BERT (Bidirectional Encoder Representations from Transformers) is a language representation model introduced by Devlin et al. [2]. Encoding text using BERT has been the standard method due to its bidirectional approach of encoding text. It is designed to pre-train deep bidirectional representations from unlabeled text, which makes it a suitable choice for text encodings. "BERT-base-uncased" is a widely used variant of BERT. The "uncased" attribute signifies its case-insensitivity, facilitating a broader understanding of language. This model excels in capturing contextual relationships between words due to its bidirectional nature and extensive pre-training on large corpora. All BERT models used in this work produce embeddings with a hidden size of 768.
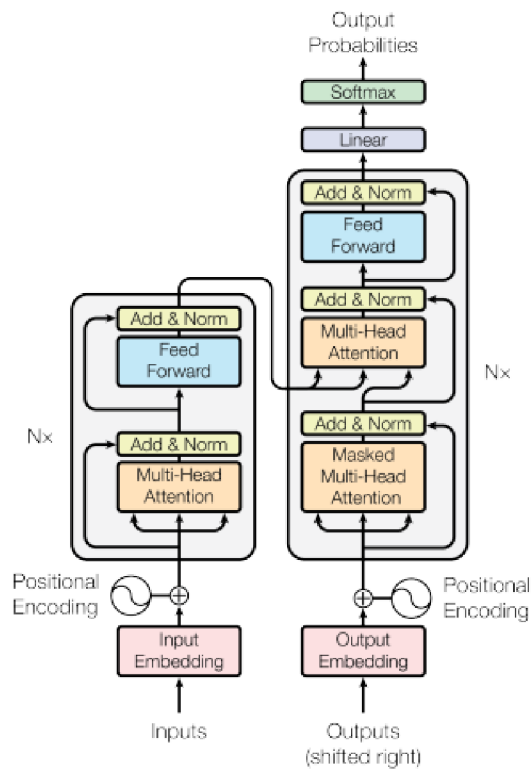
Figure 2.1: The Transformer model architecture proposed in the paper Attention is All You Need

2.1.2 mBERT

mBERT (multilingual BERT) is another variant of BERT designed to handle multiple languages. Unlike language-specific models, mBERT is trained on diverse multilingual corpora, making it proficient in understanding and generating representations for text in various languages. This model enabled researcher to work with multilingual datasets efficiently, providing a unified framework for cross-lingual NLP tasks. While this analysis is focused on English text, the inclusion of mBERT is for exploratory purposes. The aim is to showcase how a multilingual model behaves in this analysis. Additionally, its multilingual capabilities offers the flexibility to extend the analysis to other languages if needed, enhancing the adaptability and potential scope of the research.

2.1.3 TinyLlama-1.1B-Chat

Introduced by Zhang, Peiyuan, et al. [3], TinyLlama is a chat model that is trending very recently and available on Hugging Face[1], it adopts exactly the same architecture and tokenizer as Llama2 [4]. The model is compact with only 1.1B parameters, a context window of size 2048

---

[1]`https://huggingface.co/TinyLlama/TinyLlama-1.1B-Chat-v1.0`

tokens, and hidden size of 2048, which makes it possible to inference in a constrained computational setting. Compared to models like GPT 3 [5], there are few architectural differences done in Llama:

- Llama uses SwiGLU activation function instead of ReLU.

- It uses rotary positional embeddings instead of absolute positional embedding.

- It uses root-mean-squared layer-normalization instead of standard layer-normalization.

2.1.4 AMR Parser

Encoded Abstract Meaning Representation (AMR) are explored in this analysis through the use of an AMR parser which follows BART architecture. BART [6] employs a generative model with a full encoder-decoder transformer architecture, combining bidirectional and autoregressive training objectives for sequence-to-sequence models. The parser is trained on the objective of generating AMR graphs. The use of the parser is through *amrlib*[2], a python library for AMR parsing, generation and visualization. More specifically, the model 'parse-xfm-bart-base' is used to encode the data using its encoder component. The purpose of using the AMR parser is to explore whether extracting the key concepts from text can be beneficial in the task of topic modeling.

2.1.5 KeyBERT

KeyBERT is an algorithm that employs BERT embeddings for keyword extraction from texts. By leveraging contextualized word representations, it performs context-aware and accurate identification of key terms in the input text. KeyBERT utilizes a pre-trained BERT model to obtain contextualized word embeddings for each word in the input text. The word embeddings are then aggregated to produce a sentence-level embedding, this step captures the overall context of the text. Using sentence embeddings, KeyBERT identifies the most informative words as keywords. The algorithm selects words based on their contribution to the text's overall contextual meaning. In this work, KeyBERT is used to extract the most relevant words from each cluster in order to identify the relevant topic of the cluster.

2.2 K-means Clustering

K-Means is a widely used unsupervised machine learning algorithm used for clustering data points. It divides a dataset into k clusters based on similarity, aiming to minimize the intra-cluster variance. The algorithm operates by initializing cluster centroids, assigning data points to the nearest centroid, and then iteratively updating centroids based on the mean of the assigned points. An example of some random data points clustered using k-means with k set to 4 is shown in figure 2.2. The following is a detailed explanation of how k-means works.

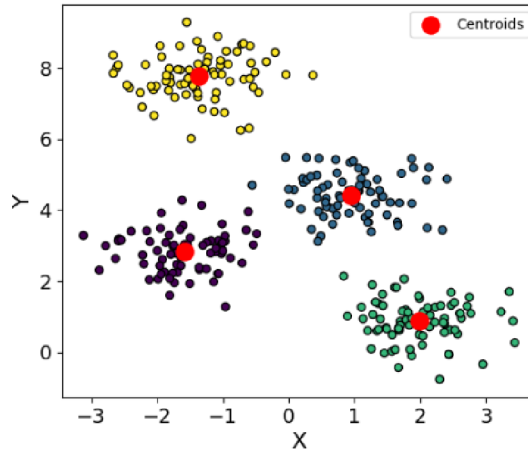
[2]https://github.com/bjascob/amrlib

Figure 2.2: Scatter plot example of clustered data using k-means with k=4

2.2.1 Initialization of Cluster Centroids

The algorithm starts by randomly initializing k cluster centroids in the feature space. These centroids represent the centers of the initial clusters.

2.2.2 Assigning Data Points to Nearest Centroids

Each data point in the dataset is assigned to the cluster whose centroid is closest to it. This assignment is based on a distance metric, commonly the Euclidean distance.

2.2.3 Iterative Update of Centroids

After the initial assignment, the algorithm iterates between two steps:

1. **Update Centroids**: For each cluster, the centroid is updated to the mean of all the data points assigned to that cluster. This moves the centroid closer to the center of the cluster.

2. **Reassign Data Points**: Once the centroids are updated, data points are reassigned to the cluster whose centroid is closest to them.

2.2.4 Minimization of Intra-cluster Variance

The goal of K-means is to minimize the within-cluster variance, also known as inertia or sum of squared distances from each point to its assigned centroid. This is achieved through the iterative process of centroid updating and point reassignment.

2.2.5 Convergence Criteria

The algorithm continues iterating until either the centroids no longer change significantly between iterations or a specified number of iterations is reached. Convergence typically occurs when the centroids stabilize and the assignment of data points to clusters remains constant.

2.2.6 Choosing the Number of Clusters (K)

Determining the appropriate number of clusters, k, is crucial. Common methods for selecting k include the silhouette analysis or elbow method. In this work, the silhouette analysis is used to determine the appropriate number of clusters for each set of encodings.

2.3 Abstract Meaning Representation (AMR) Graphs

AMR graphs are a popular way to represent the meaning of the text, abstract away from its complex terminology. It uses concepts and relations from a fixed vocabulary to capture the core semantic meaning of a sentences independently of their surface structure, in the sense that sentences which are similar in meaning should be assigned the same AMR, even if they are not identically worded. For example, the sentences "he described her as a genius", "his description of her: genius", and "she was a genius, according to his description" are all assigned the same AMR [7]. Figure 2.3 shows the graph representation of the previous sentences along with the linearized version of the AMR graph.

In this work, an AMR parser is used to encode the input text. The aim here is to explore whether encoding the text with a model that is trained on the objective of generating AMR graphs can introduce any improvements in the task of topical clustering of text.
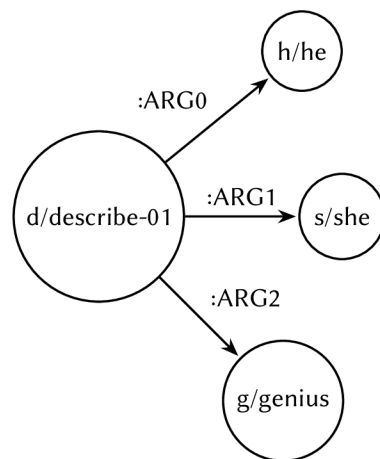

Figure 2.3: AMR graph of the Sentence "he described her as a genius".
Linearized format: (d/describe-01 :ARG0 (h/he) :ARG1 (s/she) :ARG2 (g/genius))

2.4 Evaluation Metrics

The Silhouette Score, Calinski-Harabasz Index, and Davies-Bouldin Index are metrics used to evaluate the performance of clustering algorithms. They provide quantitative measures for assessing the quality of the formed clusters. An explanation is provided of each:

2.4.1 Silhouette Score

The Silhouette Score serves as a quantitative measure of the degree of separation between clusters and the similarity of each data point within a cluster to others. It ranges from -1 to 1, where a high score indicates well-defined clusters, 0 suggests overlapping clusters, and negative values imply that data points might be assigned to the wrong cluster. The Silhouette Score is calculated using the formula:

$$\text{Silhouette Score} = \frac{1}{N} \sum_{i=1}^{N} \frac{b_i - a_i}{\max\{a_i, b_i\}}$$

where N is the total number of data points, a_i is the average distance from the i-th data point to the other data points in the same cluster, and b_i is the smallest average distance from the i-th data point to data points in a different cluster.

2.4.2 Calinski-Harabasz Index

The Calinski-Harabasz Index calculates the ratio of between-cluster variance to within-cluster variance. It measures how well-separated clusters are and is higher when clusters are compact and well-defined. It is computationally efficient and useful for datasets with varying cluster sizes. The Calinski-Harabasz Index is calculated using the formula:

$$CH = \frac{B}{W} \times \frac{N - k}{k - 1}$$

where CH is the Calinski-Harabasz Index, B is the between-cluster dispersion, W is the within-cluster dispersion, N is the total number of data points, and k is the number of clusters.

2.4.3 Davies-Bouldin Index

The Davies-Bouldin Index evaluates the compactness and separation of clusters. It measures the average similarity between each cluster and its most similar cluster, aiming for lower values, which indicate more distinct clusters. The Davies-Bouldin Index is calculated using the formula:

$$DB = \frac{1}{k} \sum_{i=1}^{k} \max_{j \neq i} \left(\frac{S_i + S_j}{d(c_i, c_j)} \right)$$

where DB is the Davies-Bouldin Index, k is the number of clusters, S_i is the average distance from each point in cluster i to the centroid of cluster i, and $d(c_i, c_j)$ is the distance between the centroids of clusters i and j.

2.4.4 Visual Example

A visual example in Figure 2.4 demonstrates how clustering scores correspond to increasing the value of k. In this example, the data clearly fit into two clusters. Attempting to fit the data into more than two clusters results in lower Silhouette and Calinski-Harabasz scores and higher

Davies-Bouldin scores. This serves as a good illustration of how these scores can indicate the appropriate number of clusters that best fit the clustered data. In this study, the Silhouette score will be employed as a method to determine the most suitable value of k for each of the encodings obtained by the different models. The k-means algorithm will be executed multiple times using various values of k, and the final choice of k will be based on the Silhouette score that best aligns with the clustering quality.
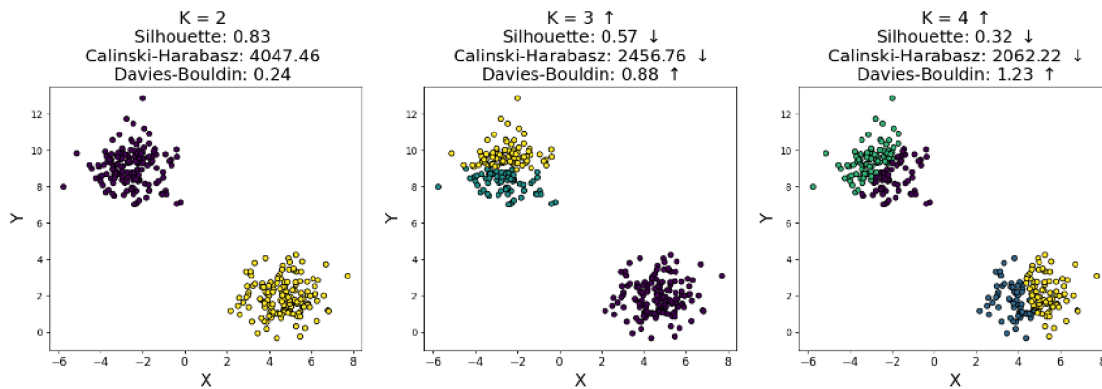

Figure 2.4: Visual example of how clustering scores correspond to increasing the number of clusters (k) beyond the suitable value, which is 2 in this example

2.5 Autoencoders

An autoencoder is a type of artificial neural network used to learn efficient codings of unlabeled data (unsupervised learning). An autoencoder learns two functions: an encoding function that transforms the input data, and a decoding function that recreates the input data from the encoded representation. The autoencoder learns an efficient representation (encoding) for a set of data, typically for dimensionality reduction. Figure 2.5 shows the basic architecture of the autoencoder. Deep autoencoders are employed in this work in the process of encoding the embeddings into a lower dimensional space after finding the least possible number of components that can recreate the embeddings with minimum recreation loss. This is going to be helpful for k-means since the reduced dimensional space boosts the performance of the clustering algorithm.

2.6 UMAP for Dimensionality Reduction

UMAP (Uniform Manifold Approximation and Projection) is a dimensionality reduction technique designed for visualizing high-dimensional data in a lower-dimensional space. It preserves the underlying manifold structure of the data, maintaining meaningful relationships between points. UMAP is very good at preserving both local and global structure, offering flexibility in parameter tuning and scalability. It has applications in visualization, clustering, and dimensionality reduction tasks, providing insights into complex datasets. Compared to other
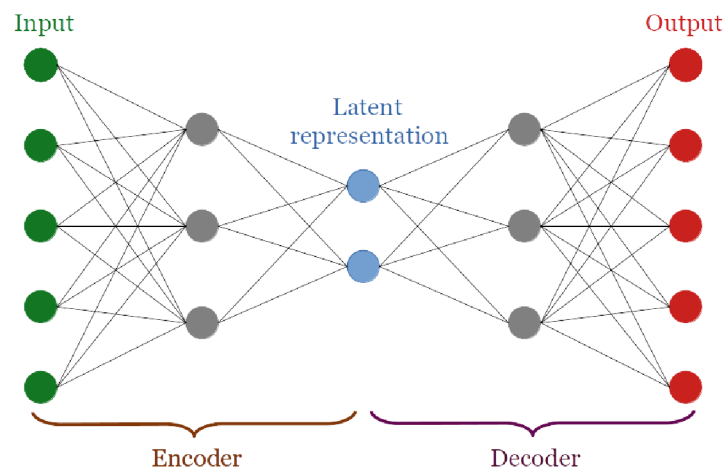
Figure 2.5: Deep Autoencoder

techniques like t-SNE (t-Distributed Stochastic Neighbor Embedding) and PCA (Principal Component Analysis), UMAP stands out for its effectiveness in capturing nonlinear relationships and handling large datasets efficiently.

3 Related Work

Recent years have witnessed the widespread adoption of transformer models in various Natural Language Processing applications, resulting in significant improvements across several NLP tasks. Numerous studies have proposed effective methods to leverage contextualized transformer embeddings for downstream applications. This chapter explores relevant previous research that has influenced the foundation of this thesis and highlights its distinctions.

In the task of topic modeling, Zineddine Bettouche and Prof. Andreas Fischer [8] conducted topical clustering of scientific research activity within DIT's publication library, utilizing the same dataset as employed in this work. The authors employed various models from the BERT family, including BERT-base, SciBERT, as well as mBERT for processing multilingual text. Their research involves creating a landscape representation of scientific fields through encoding and clustering research publications. As ground-truth topic labels are absent, coauthorship analysis is employed, examining author uniqueness within clusters and constructing coauthorship-based social networks. The calculated high uniqueness of authors in the formed clusters and the found homogeneity of topics across the connected-components in the social network is used in assessing the effectiveness of the clustering. The coauthorship analysis part is not conducted in this thesis work. However this work extends their approach by incorporating a chat model encodings and a set of BERT encodings reduced to a lower dimensional space using autoencoders. Additionally, encoded Abstract Meaning Representation graph information in the context of topic modeling is explored.

Another study [9] introduces an unsupervised topic detection approach to address the challenge of discovering current research topics and methodologies in scientific domains of a number of publications. Leveraging transformer-based GPT-3 similarity embedding models and modern document clustering techniques, the approach is demonstrated using 593 publication abstracts from urban study and machine learning domains. The process involves three phases: an iterative clustering phase utilizing GPT-3 embeddings and HDBSCAN clustering to group similar abstracts, a keyword extraction phase employing the Maximal Marginal Relevance ranking algorithm, and a keyword grouping phase producing topic representations for abstract clusters. The authors used Uniform Manifold Approximation and Projection (UMAP) algorithm as a dimensionality reduction technique to be able to reduce the high-dimension space of the abstract embeddings to a reasonable range because HDBSCAN requires the dimension size to be smaller than the number of abstracts, in their case, smaller than 593. However, in this work, autoencoders are used to encode the embeddings to an appropriate space based on minimized reconstruction-loss, while UMAP is only employed for the visualization purpose.

This paper [10] explores the application of the Bidirectional Encoder Representation from Transformers (BERT) model for text clustering, comparing it with the commonly used Term

Frequency Inverse Document Frequency (TFIDF) method. Their results indicate that BERT outperforms TFIDF in 28 out of 36 metrics, highlighting its effectiveness in representing textual data for clustering. Additionally, the paper emphasizes the importance of adapting feature extraction and normalization techniques based on the chosen text clustering algorithm.

A study [11] investigated the utilization of pre-trained Transformer-based word embeddings in the context of text clustering. The authors introduce a clustering ensemble approach that incorporates embeddings from all layers of the network. Numerical experiments conducted on datasets with various Transformer models demonstrate the effectiveness of the proposed method when compared to several baseline methods.

Prior research work explored using encoded AMR graph information. In a recent study by Joseph Gatto and Sarah M. Preum [12], AMR graphs were leveraged to model low-resource health Natural Language Processing tasks. Through the augmentation of text embeddings with semantic graph embeddings, the authors demonstrated improved performance across six classification tasks. Inspired by their work, this study explores whether encoding AMR information can yield any interesting findings. The process of linearizing Abstract Meaning Representation (AMR) structures enables the application of traditional sentence embedding techniques, such as contrastive learning, to construct meaningful AMR representation vectors. Contrastive learning involves creating a dataset of triplets comprising an anchor, a positive example, and a negative example. The objective is to encourage the model to bring the embeddings of the anchor and positive example closer while pushing the anchor and negative example embeddings further apart. This approach yields semantically rich text embeddings that are analyzable in high-dimensional space. Unlike Gatto et al.'s approach, this work involves encoding AMR information using an AMR parser, specifically focusing on the encodings produced by the parser's encoder component.

4 Methodology

This chapter focuses on the detailed steps of the implementation process. In this chapter the focus purely on the methods, while the next chapter will show the results of the experimentations.

4.1 Dataset Exploration

The scientific publication database at DIT consists of a total 1500 publications that include an abstract section. To ensure consistency in language, 175 entries identified as non-English are excluded from the dataset, resulting in a remaining 1325 publications for analysis. The language filtering process utilizes *langdetect*, a library designed to detect the language of a given text. Despite this automated approach, a few German papers were identified during manual verification, resulting in manual exclusion for those cases.

With regards to the length of the analyzed abstracts, Figure 4.1 shows a histogram illustrating the frequency of the number of tokens. This is a crucial consideration due to context length restrictions imposed by Transformer models. However, the majority of abstracts fall within an acceptable range, with an average token count of 189 tokens. BERT models have a context length limitation of 512 tokens, if the token count crosses this threshold, truncation occurs, resulting in the loss of valuable information. In the case of TinyLlama, this concern is alleviated, as its context window length is 2048 tokens, ensuring that no truncation will happen since all analyzed abstracts are already below that limit.
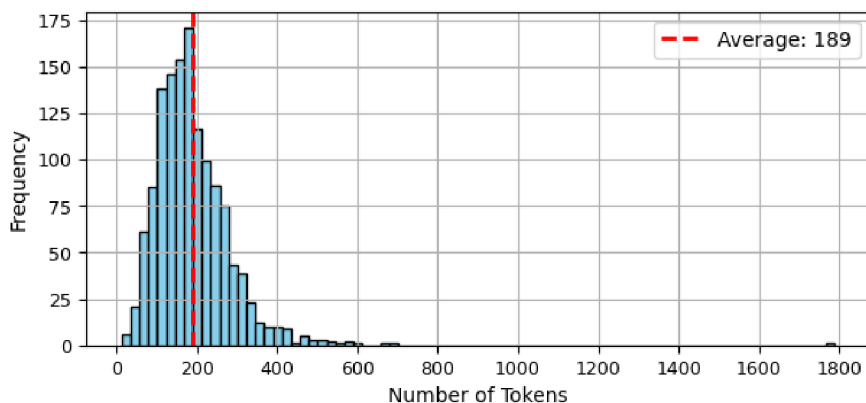

Figure 4.1: Histogram of token count per abstract

4.2 General Overview

While the used models are different in their architecture and learning objectives, the steps are generally similar. Figure 4.2 shows a diagram that illustrates the overall process applied in this work.
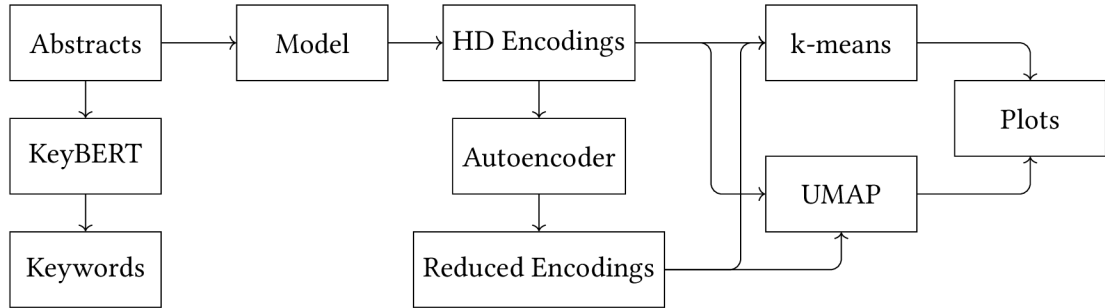

Figure 4.2: Methodology Overview

The abstracts are fed to BERT-base, mBERT, TinyLlama, and the AMR Parser to obtain the encodings, as well as to KeyBERT in order to obtain the most relevant keywords from each abstract. The acquired information is then stored for the next stage, which is the clustering using k-means and visualization using UMAP. In addition to the clustering and visualization of the high-dimensional encodings, a reduced set of encodings is explored as well. The choice of the most suitable set of embeddings to be reduced is decided based on the Autoencoder Reconstruction-Loss Assessment. The following subsections describe the process in more details.

4.2.1 Encoding of Scientific Publications

To obtain the contextualized text embeddings of DIT's scientific publication database, each model is fed with the abstract of each publication. The output of the last encoder layer is of particular interest; it comprises a sequence of vectors, with each vector representing an input token in its surrounding context. Following this, an average vector is calculated from the word-level embeddings, resulting in a single vector serving as the overall sentence embedding. Figure 4.3 that illustrates this process.
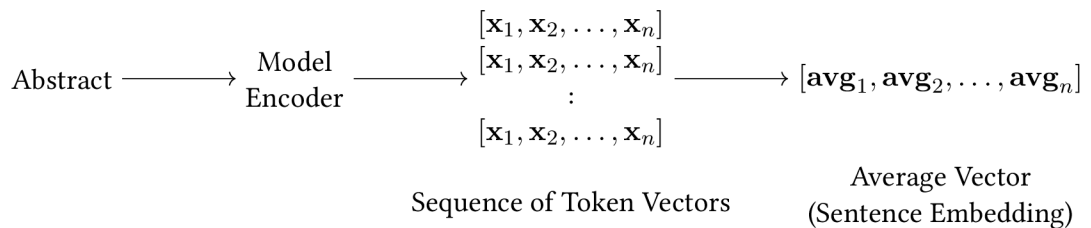

Figure 4.3: Encoding of Scientific Publications

4.2.2 Clustering and Plotting of The Encoded Scientific Publications

Following the inference of each model and the storage of the resulting embeddings, the next step involves applying the k-means clustering algorithm. The number of clusters k will vary based on the nature of the embeddings and their ability to effectively group data points. Instead of relying on intuition in choosing k, the value that yields the best possible Silhouette score for each set of embeddings will be selected. To achieve this, each set of embeddings is clustered multiple times with k ranging from 2 to 20. Afterwards, the optimal value for each case is determined.

For the visualization purpose, the dimensionality of the clustered embeddings needs to be reduced to a 2-dimensional space suitable for plotting. The python library *umap-learn*[1] is used for the dimensionality reduction task, while *matplotlib*[2], a well-known Python library, will be employed for the visualization part.

4.2.3 Keyword Extraction

In order to determine the primary theme or field of study characterizing the content of each cluster, KeyBERT will be employed to extract the top 3 relevant keywords from each abstract. Following the clustering of abstracts, the most frequent words in each cluster can be computed. A number of 4 to 6 keywords are sufficient to identify the relevant topic of a cluster. These keywords are then displayed in the legend of each plot, along with their corresponding topics.

4.2.4 Determining Topics

Assigning a relevant topic to each cluster primarily involves analyzing the top frequent keywords within the cluster. In addition to common sense and manual checks on the content of each cluster, arriving to the final topic is also done in consultation with a generative model. ChatGPT is a suitable option for this task since this is only for confirmation and no scalable solution is required here. It has the capability to recognize the main topic or field of study based on a given list of keywords. The prompting method can be controlled to decide whether ChatGPT should provide a general topic characterized by the keywords, or a more specific one. For example, keywords like "classification," "algorithm," "prediction," and "kernel" may result in the general topic of Computer Science, but on a more detailed level, it can also be "Machine Learning". Table 4.1 presents lists of possible keywords and their corresponding topics. This predefined list is utilized to automatically assign the field of study, which is then displayed in each plot's legend. The topic is assigned to a cluster when its top 4 most frequent keywords overlap with at least 3 words from the list.

4.2.5 Autoencoders for Dimensionality Reduction

This work includes the training and utilization of autoencoders to effectively reduce the dimensional space of the encoded abstracts. This reduction aims to boost the performance of k-means and improve the separation of topics. It is known that any dimensionality reduction

---

[1] `https://umap-learn.readthedocs.io/en/latest/`
[2] `https://matplotlib.org/`

Keywords	Topic
recommender, recommendations, personalized, analytics	Big Data
rna, cell, cells, melanoma, myelin, melanoma	Biology
tourist, cultural, motivation, satisfaction	Crowdsourcing
digital, biomimetics, mobile, data, health, participatory	Digital Health
laser, dielectric, microscopy, capacitors, silicon, ghz, electrode	Electronics
renewable, electricity, energy, solar	Energy
biopsy, therapy, fmri, health, training, exercise, overweight, lifestyle, patients, aerobic, metabolic	Health
network, virtual, virtualization, security, digital	IT
classification, algorithms, features, clustering, prediction, kernel	Machine Learning
quantum, exciton, silicon, photoluminescence, irradiation, manufacturing, calibration	Materials Science
quality, video, 3d, visual, bitrate, bitstream, mpeg	Media
reactor, thermal, coolant, reactors, nuclear, fusion, bubbles	Molecular Physics
nanowire, nanoporous, microlenses, micro, laser, plasma	Nanotechnology
polishing, optical, optics, surfaces, surface, laser, precision	Optical Engineering
piezoelectric, actuators, buckling, stability, piezoelectric	Structural Engineering

Table 4.1: Topic assignment based on frequent keywords

technique involves a degree of information loss. To minimize this loss, it is important to carefully select the optimal number of components that can still retain the internal structure of the data. This is achieved through the Autoencoder Reconstruction-Loss Assessment which measures the reconstruction loss while having different number of components in the latent space of the autoencoder, and then encode the embeddings into a lower space after deciding on the appropriate latent space size.

Autoencoder Reconstruction-Loss Assessment

The approach taken involves training multiple autoencoders for each set of embeddings generated by the models. These autoencoders vary in the number of components on the latent space (encoder output layer), ranging from 1 component up to 140 with a step of 10. Each autoencoder consists of input and output layers adapting to the specific embeddings' hidden size, along with five hidden layers, including the latent space layer. The hidden layer sizes are set as 256, 128, X, 128, 256, where X represents the encoded latent space which is assigned a different value in each iteration. The purpose of this process is to assess the validation loss for each autoencoder and determine the optimal number of components that minimize information loss and retain the essential structure of the original embeddings.

Encoding The Embeddings Into A Lower Space

Following the autoencoder reconstruction-loss assessment and the determination of the optimal number of components for the encoder latent space, the next step involves encoding the original embeddings into this lower-dimensional latent space. This is accomplished using a deep autoencoder comprising 7 hidden layers set as 512, 256, 128, X_{optimal}, 128, 256, 512, where X_{optimal} is the previously identified optimal value. The resulting reduced set of embeddings is then subjected to clustering and plotted in a similar approach applied to the original embeddings.

5 Experiments

This chapter reveals the results of the experiments. To present the findings in an organized manner, the clustered encodings in their original high-dimensional space are shown first. Following that, the advantages of encoding a set of embeddings into a lower space through the utilization of autoencoders is explored.

5.1 Original Embeddings Findings

This first section is going to show the findings of the embedded abstracts in their original high-dimensional space, without any dimensionality reduction applied prior to k-means clustering. The results begin by conducting the Silhouette score assessment where the optimal number of clusters is decided for each set of embeddings. After that, the clustering results of BERT-base, mBERT, TinyLlama, and AMR parser are presented.

5.1.1 Silhouette Score Assessment

As discussed in the Methodology chapter, the most suitable value for the number of clusters k is the one that results in a high Silhouette score. Figure 5.1 shows the Silhouette score assessment result. It is important to highlight that the selection of the optimal value is not solely based on the highest score but involves consideration of the entire plot shape.
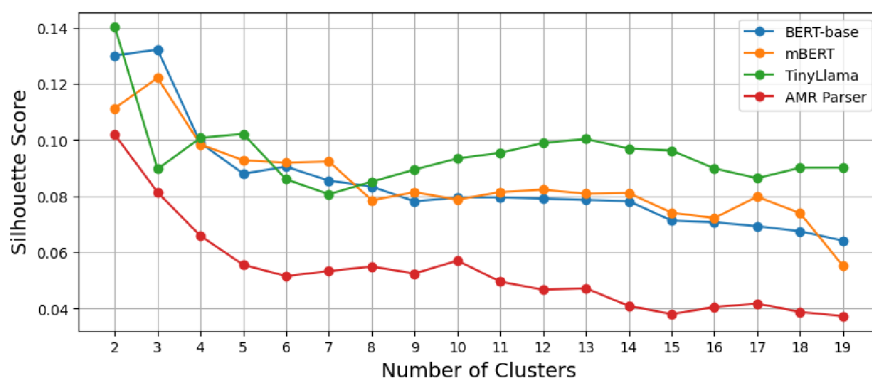

Figure 5.1: Silhouette Score Assessment

Based on the assessment, k values of 8, 7, 14, and 6 for BERT-base, mBERT, TinyLlama, and the AMR Parser, respectively, are chosen. This approach ensures a fair comparison, as each model is evaluated based on the most suitable number of clusters for its specific characteristics.

5.1.2 BERT-base-uncased

Figure 5.2 presents the clustering results of the transformer-encoded scientific publications using the model BERT-base-uncased. It is possible to discover 8 different topics with BERT-base. In terms of cluster quality, BERT-base achieves Silhouette, Calinski-Harabasz, and Davies-Bouldin scores of 0.083, 87.036, and 2.593 respectively. The identified topics are Energy, Material Science, Health, Electronics, IT, Media, Optical Engineering, and Biology. Looking at the most frequent keywords for each cluster shows confidence in the topic assignment, which indicates an appropriate choice of 8 clusters.
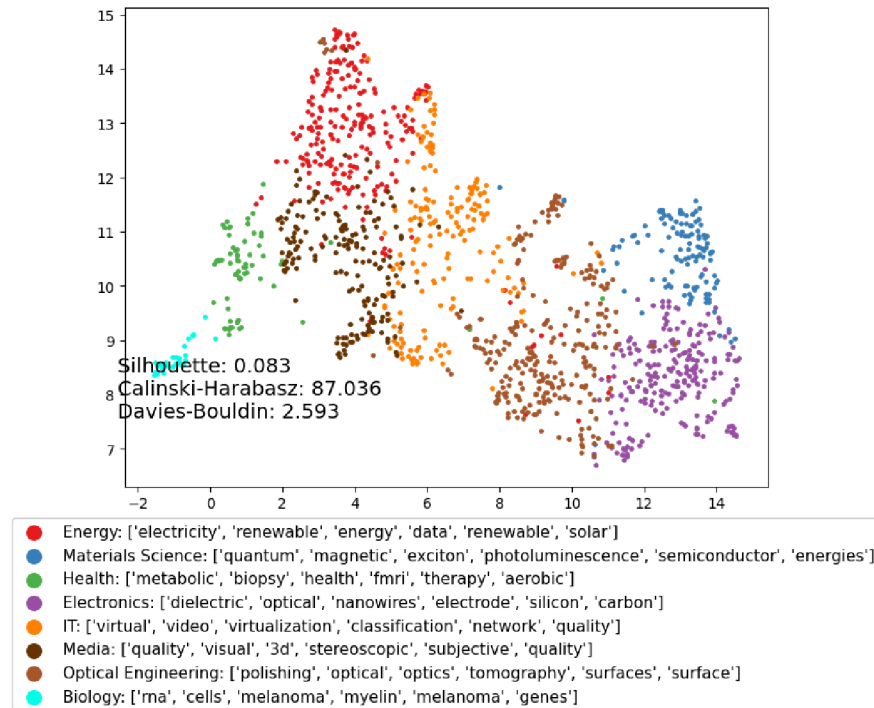

Figure 5.2: BERT-base-uncased Encodings

5.1.3 mBERT

Figure 5.3 shows the clustering results of mBERT encodings, it is possible to discover 7 clusters with mBERT. It achieves Silhouette, Calinski-Harabasz, and Davies-Bouldin scores of 0.093, 88.243, and 2.772 respectively. BERT-base, in comparison, was capable of recognizing one additional cluster "IT", which probably most of its content got embedded with the "Media" cluster, considering their high similarity.

An argument favoring BERT-base over mBERT for this specific analysis is that the input text is entirely in English, giving BERT-base an advantage with more exposure to English training samples compared to mBERT, which was trained on 104 different languages. An interesting observation in the mBERT plot is the distinct green cluster on the left, labeled as "Health." This cluster contains mainly health-related publications, along with some publications in the Energy

and a few other topics, as indicated by the keywords "renewable" and "dielectric". While there is no clear explanation for this behavior at the moment, it raises a question mark that could be investigated in future analyses involving multilingual data.
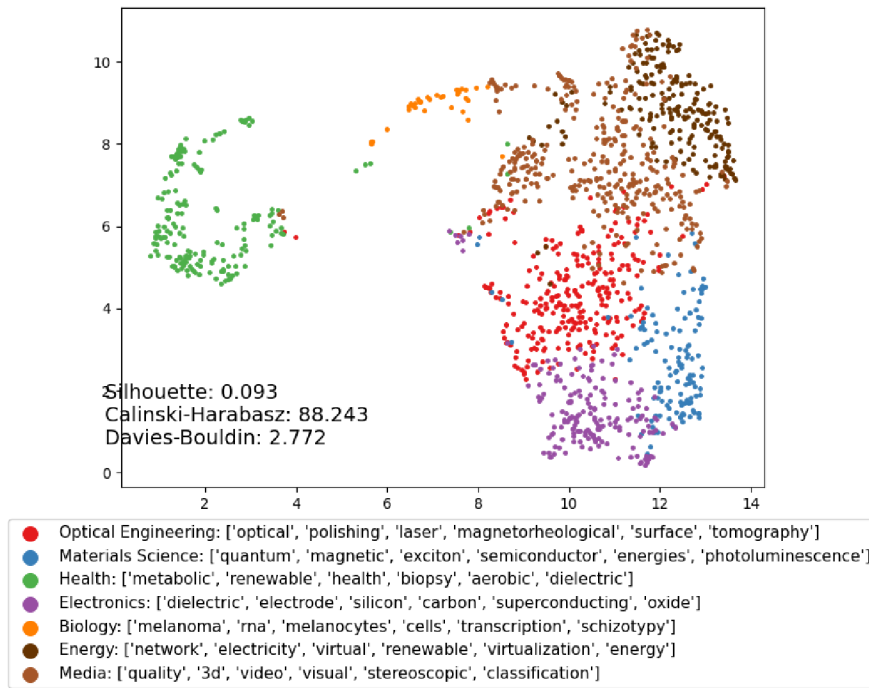

Figure 5.3: mBERT Encodings

5.1.4 TinyLlama-1.1B-Chat

Figure 5.4 shows the results of clustering the scientific publications using the model TinyLlama-1.1B-Chat. The results were comparable to BERT-base in terms of clustering scores. TinyLlama achieves Silhouette, Calinski-Harabasz, and Davies-Bouldin scores of 0.098, 64.815, and 2.532 respectively. However, the number of identified clusters is 14, which is significantly higher than the BERT models. An important consideration here is that TinyLlama encodes the input text into a 2048-dimensional space, while BERT encodings are in 768 dimensions. Despite the potential disadvantage of the significantly high dimensionality in TinyLlama (almost 2.6 times higher than BERT), exposing it to the Curse of Dimensionality effect, it still achieved comparable clustering scores, and that certainly counts for TinyLlama.

With TinyLlama, it was possible to discover interesting detailed topics, such as:

- Digital Health: A discipline that includes digital care programs, integrating technologies and health, and analyzing public health.

- Crowdsourcing: A topic that involve studying social and cultural topics of a group of people.
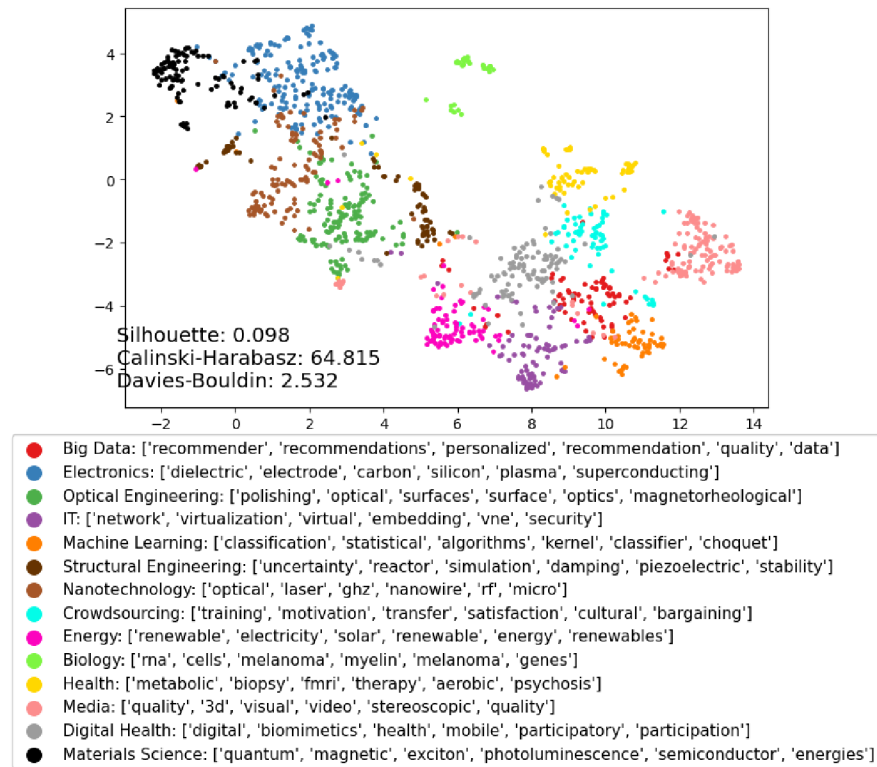
Figure 5.4: TinyLlama-1.1B-Chat Encodings

- Big Data: Involves analyzing large amounts of data, as seen in recommender systems.

- Structural Engineering: A branch of civil engineering that focuses on the design and analysis of structures such as buildings, reactors, and other infrastructure.

To grasp the overall shift in the distribution of abstracts across topics between TinyLlama and BERT-base, a visual representation of the mapping between the two clustering methods is provided using the sankey diagram in Figure 5.5. The diagram illustrates that the diversity in TinyLlama's technology-related topics primarily originates from what was initially assigned to "IT," "Media," and "Energy" clusters according to BERT-base. Additionally, TinyLlama successfully separated the "Nanotechnology" from "Electronics", as well as "Structural Engineering" topic from what was mistakenly included with "Optical Engineering." The rest of topics, namely "Material Science," "Health," and "Biology," generally remain consistent with minor differences.
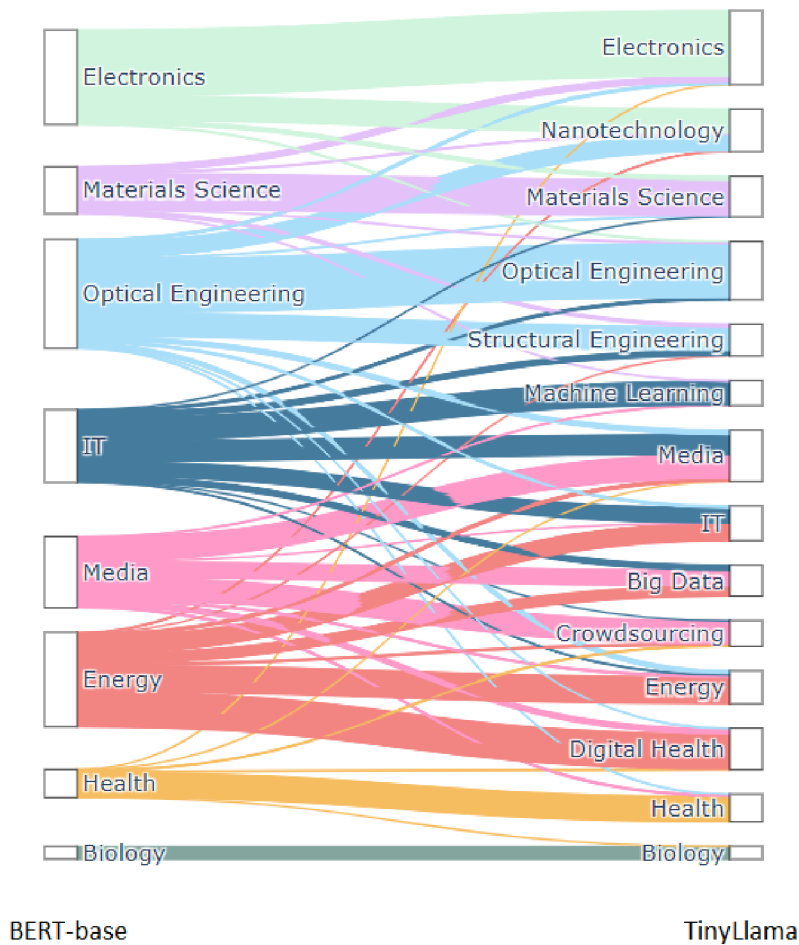
Figure 5.5: Sankey diagram for topic assignment comparison of BERT-base and TinyLlama

These interesting findings demonstrate the effectiveness of the chat model TinyLlama in the task of topic modeling, outperforming BERT-base in this specific task, due to its capability of revealing the diversity of topics covered within the DIT publication database while achieving comparable clustering scores. It is also a motivation to extend this work and include larger variations of Llama2 in future analyses.

5.1.5 AMR Parser

The result of clustering the encodings coming from the AMR parser did not perform as good as the previous models in the task of topic modeling. The parser achieves Silhouette, Calinski-Harabasz, and Davies-Bouldin scores of 0.052, 77.416, and 2.901 respectively. It is clear that some data points were assigned the wrong cluster, and it is possible to confirm that by looking at the "Health" cluster where keywords like "renewable" and "electricity" should not appear

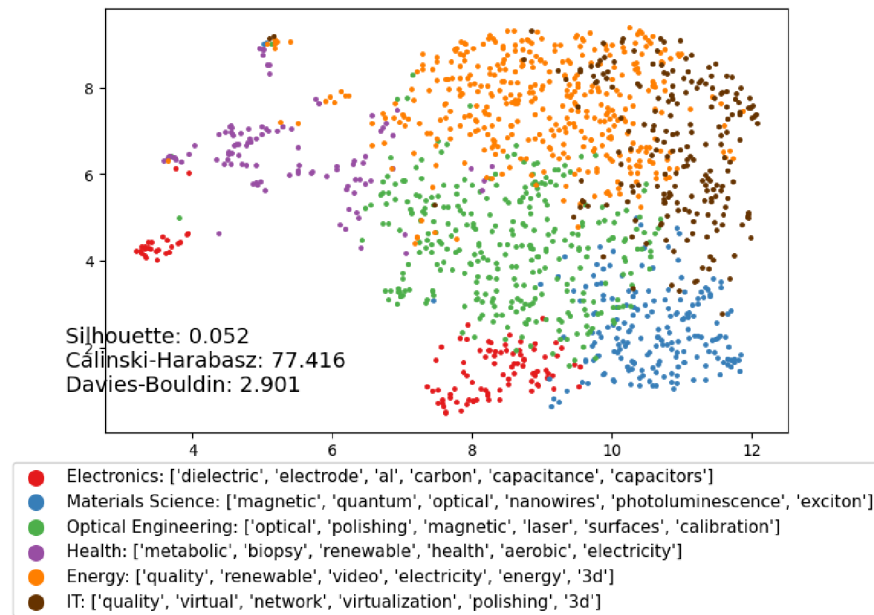there, at least within the most frequent keywords for the cluster.

Figure 5.6: AMR Parser Encodings

It is possible that AMR's tendency to generalize concepts lead to encodings being closer to each other, making it harder to distinguish detailed topics. As a result, this AMR encodings didn't turn out to be a good fit for the task at hand.

5.2 Reduced Embeddings Findings

This section explores the advantages of encoding the embeddings into a lower-dimensional space using autoencoders, followed by clustering the reduced embeddings and presenting them in a manner similar to the previous approach. The primary objective is to enhance the performance of the clustering algorithm and assess whether improved clustering results can be achieved. Opting for the appropriate set of embeddings and latent size is achieved in two steps:

1. **Autoencoder Reconstruction-Loss Assessment:** Determining the optimal number of components that still preserves essential information from the original embeddings is crucial. This will be accomplished through the Autoencoder Reconstruction-Loss Assessment, which aids in selecting the appropriate set of embeddings and determining the number of components of the latent space of the autoencoder.

2. **Training and Validation Losses:** The training and validation losses of the autoencoder responsible for the dimensionality reduction process are examined. This step ensures that overfitting is not occurring. If the autoencoder exhibits signs of overfitting the training

set while reconstructing the embeddings, it will be excluded from the reduction process. This ensures the effectiveness of the dimensionality reduction technique employed in the clustering process.

5.2.1 Autoencoder Reconstruction-Loss Assessment

Figure 5.7 illustrates the reconstruction validation losses of the trained autoencoders across the number of encoded components. The analysis reveals that BERT-base excels in reconstructing the input with minimal loss, followed by TinyLlama. While mBERT shows comparable results, it is going to be excluded from the next experiment since BERT-base and TinyLlama are expected to be reliable options. On the other hand, the AMR parser shows the highest reconstruction errors, with an unexpected behavior after 60 components. Therefore, it makes sense to exclude the AMR parser from the reduction process.
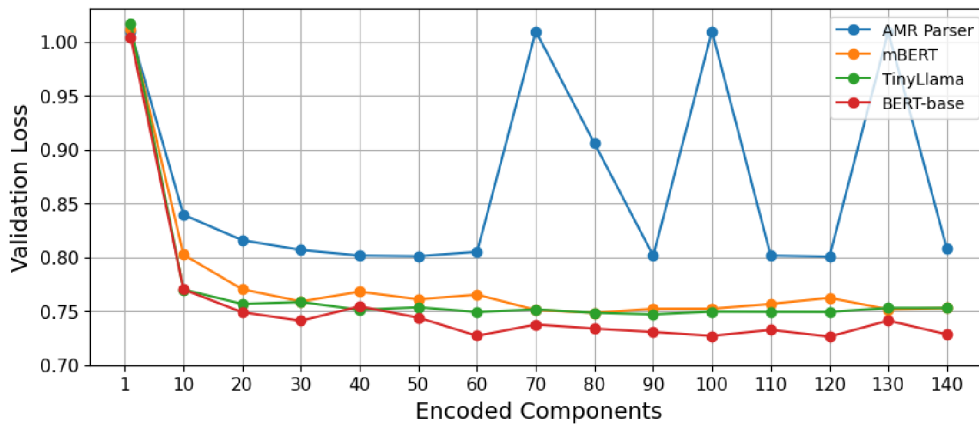

Figure 5.7: Autoencoder Reconstruction-Loss Assessment

Based on the assessment, the encodings of BERT-base and TinyLlama are chosen to move to the next round, selecting a value of 60 components for both cases. This choice ensures minimum reconstruction errors of 0.73 and 0.75 for BERT-base and TinyLlama respectively.

5.2.2 Autoencoder Training and Validation Loss

Another necessary step in opting for the appropriate set of embeddings to be encoded into a lower space is the training and validation loss of the autoencoder responsible for the reduction process. Figure 5.8 shows the training and validation loss when training two different autoencoders to reconstruct BERT-base and TinyLlama encodings.

In the case of BERT-base, the training and validation loss curves exhibit remarkable similarity, indicating a high capability to perform the task with a strong generalization to new data. On the other hand, TinyLlama is showing a significant gap between the training and validation losses, this could be a sign of overfitting because its encodings may be too complex and fitting noise in the training data. Despite TinyLlama's prior success in clustering, it is going to be

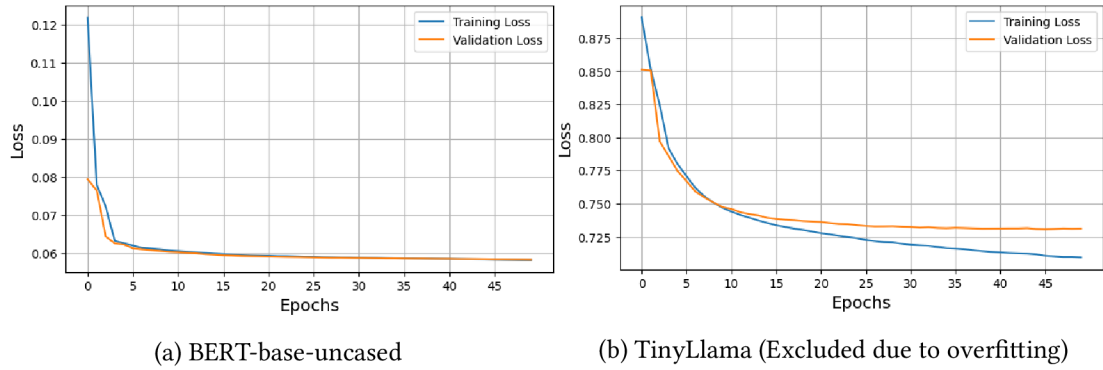| (a) BERT-base-uncased | (b) TinyLlama (Excluded due to overfitting) |

Figure 5.8: Autoencoder Training and Validation Loss

excluded from the reduction process due to the overfitting issue, and BERT-base encodings are the final choice to proceed with the reduction task.

5.2.3 Silhouette Score Assessment

In a similar sense as seen previously, the appropriate number of clusters for the reduced set of embeddings is decided based on the Silhouette score assessment. Figure 5.9 shows the assessment for the BERT-base encodings reduced to 60 components. A k value of 13 clusters is selected as it yields a silhouette score of 0.182, being the highest Silhouette score achieved in this analysis.
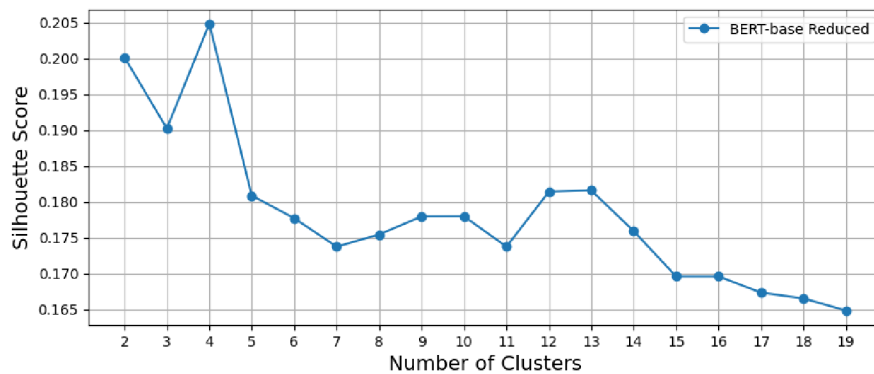

Figure 5.9: Silhouette Score Assessment (Reduced BERT-base)

5.2.4 Reduced BERT-base Encodings

Finally, the clustering outcomes of the encoded publication using BERT-base encodings after reducing to 60 components are shown in Figure 5.10. The reduced set of embeddings resulted in the best clustering scores in this analysis, achieving Silhouette, Calinski-Harabasz, and Davies-Bouldin scores of 0.18, 205.249, and 1.597 respectively.
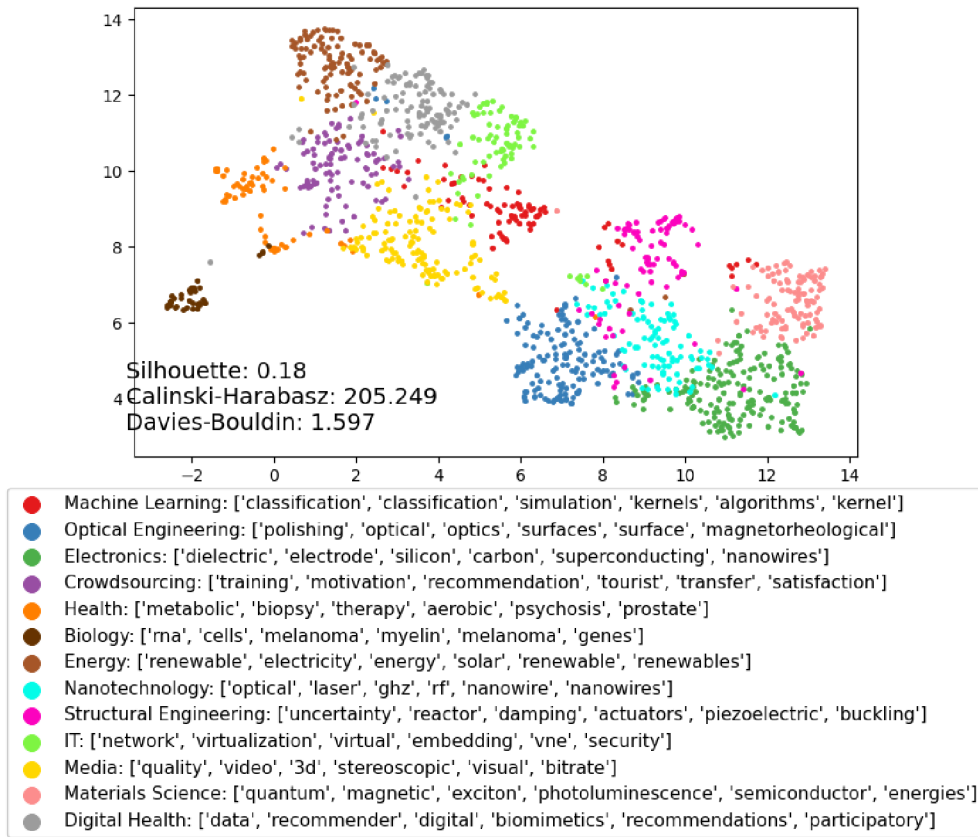
Figure 5.10: BERT-base Encodings Reduced to 60 Components

The 13 identified topics with the reduced BERT-base encodings align closely with the 14 topics discovered with TinyLlama, with the exception of the additional "Big Data" cluster found by TinyLlama. This illustrates the efficiency of encoding BERT embeddings into a lower space of 60 components. The reduced encodings demonstrate the capability to identify nearly identical topics while notably enhancing the clustering scores. The next section provides a comparative analysis of the two methods (TinyLlama and the reduced BERT-base encodings).

5.3 TinyLlama vs Reduced BERT-base Comparison

The final piece of analysis in this work presents a comparison between the two high achieving methods in this study. Up to this point, both TinyLlama and the reduced set of BERT-base encodings have demonstrated outstanding results. Figure 5.11 presents a bar chart, illustrating differences in cluster volumes between the two methods, with the difference in cluster size displayed as data labels. Despite minor variations in cluster sizes, the two methods exhibit a remarkably similar distribution of abstracts across topics.
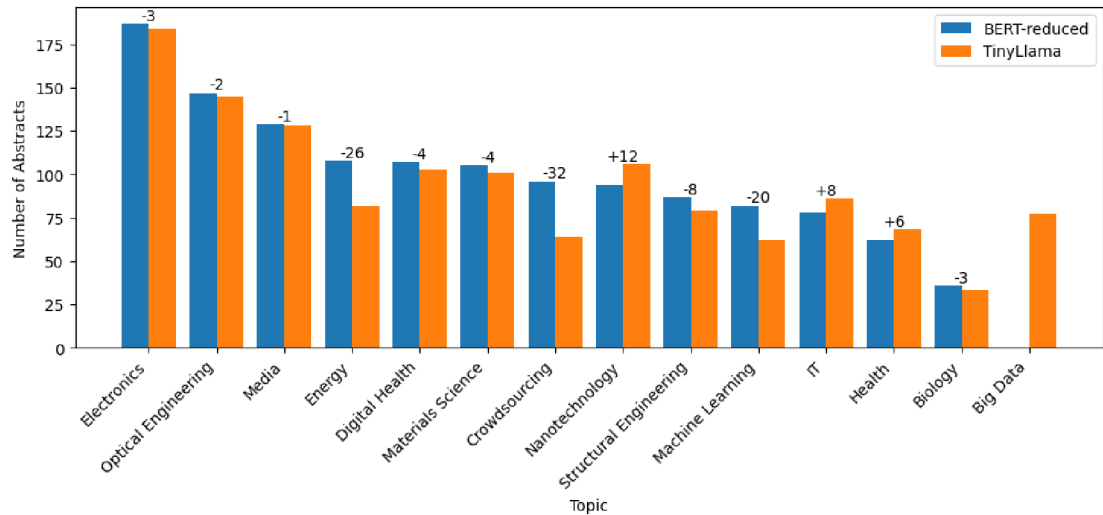
Figure 5.11: Reduced BERT vs TinyLlama topic distribution

For a more insightful perspective, a comparison between the findings of the two methods using the Sankey diagram is provided in Figure 5.12. The diagram illustrates how topics from the reduced set of BERT encodings correspond to the topics obtained from TinyLlama. The flow shows high consistency in topic assignment between the two methods, except of a few minor differences which is expected.

It is important to be aware that certain abstracts may fit to more than one suitable topic. For instance, a scientific publication within the energy sector utilizing big data or machine learning techniques might be considered as "Energy," "Big Data," or "Machine Learning" based on the semantics employed in its abstract text. Additionally, differences in specific encoding characteristics and the k-means random initialization of centroids contribute to variations in cluster assignment between any two methods. However, it's important to emphasize that such differences don't necessarily mean that one of them is completely wrong. The key insight from this comparison is that the overall distribution is largely similar in both methods.
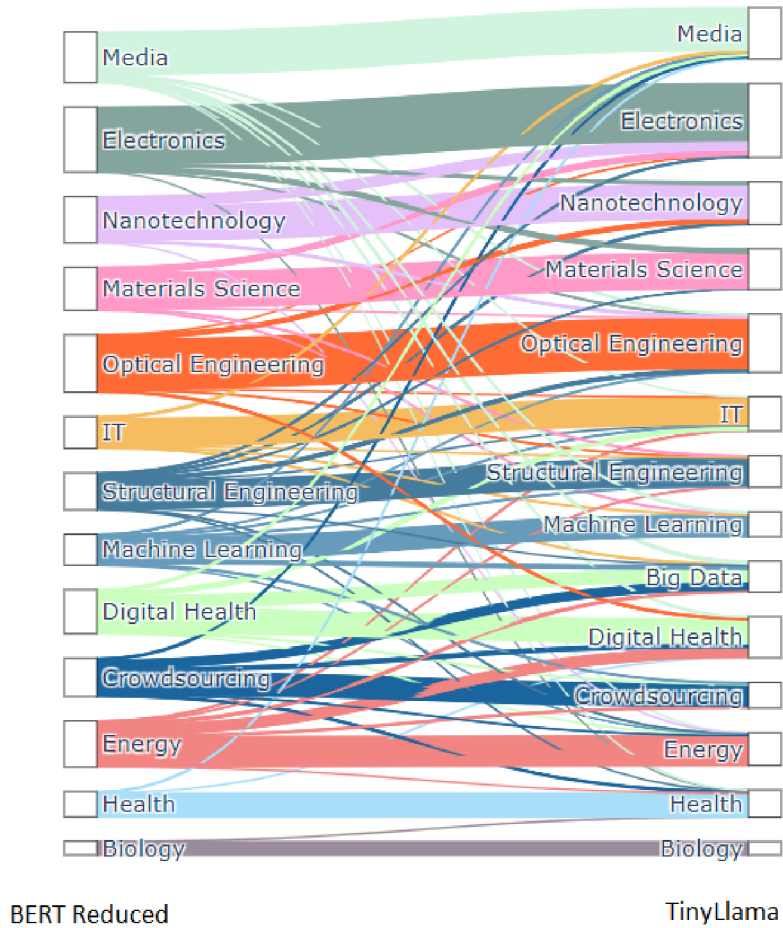
Figure 5.12: Sankey diagram for topic assignment comparison of Reduced BERT and TinyLlama

6 Results and Discussion

During this analysis, transformer encoded research publications were utilized to perform topic modeling which aims to group these documents into meaningful clusters based on their main topic. Table 6.1 summarizes the performances of the used models. It includes the number of clusters or topics identified as well as the clustering scores for all of the methods.

Model	Clusters	Silhouette	Calinski-Harabasz	Davies-Bouldin
BERT-base-uncased	8	0.083	87.036	2.593
mBERT	7	0.093	88.243	2.772
TinyLlama-1.1B-Chat	**14**	0.098	64.815	2.532
AMR Parser	6	0.052	77.416	2.901
Reduced BERT (60 components)	13	**0.18**	**205.25**	**1.597**

Table 6.1: Clustering scores of the different models

The chat model TinyLlama as well as the BERT-base encodings reduced to 60 components, have demonstrated interesting results, being able to group the documents into more detailed clusters and achieving better clustering scores compared to the base-line method using original BERT encodings. TinyLlama's encodings were able to form 14 clusters representing different topics, while the reduced BERT-base encodings formed 13 clusters. TinyLlama achieved clustering scores comparable to those obtained with original BERT-base encodings. On the other hand, the reduced BERT-base encodings achieved significant improvements in clustering scores, improving Silhouette, Calinski-Harabasz, and Davies-Bouldin scores by 116%, 135%, and 38% respectively, compared to the common BERT-base method of using high dimensional embeddings. On the other hand, the AMR parser was not a suitable option for this specific analysis since it did not introduce reliable results due to AMRs tendency in generalizing concepts, resulting in encodings being closer to each other, which then result in less clustering efficiency. The model mBERT showed some confusion in its results, grouping Health and Energy-related publications in one isolated cluster. However, mBERT could serve as a suitable option, particularly when processing multilingual data, as demonstrated in previous works.

Based on these findings, it is clear that TinyLlama and the reduced set of BERT encodings yield favorable outcomes. While the reduced BERT encodings demonstrate superior clustering scores compared to TinyLlama, it is noteworthy that TinyLlama excels in identifying an additional cluster.

7 Conclusion

This thesis conducted topical clustering on a library of scientific publications, aiming to cluster and group these publications into their main topics. The focus was on the publication database at the Deggendorf Institute of Technology (DIT), which contained 1325 scientific publications that include an abstract section written in English.

Previous methods mainly involved the use of the BERT family of models in encoding texts, being the base-line method for such tasks. However, and in addition to the traditional method using BERT, this work extended the experiments with the use of a the chat model TinyLlama-1.1B-Chat, which adopts the architecture of Llama2, as well as encoded Abstract Meaning Representation information through the use of an AMR parser. The work also explored the possible advantages of reducing the dimensional space of BERT encodings aiming to boost the performance of k-means clustering algorithm through the use of autoencoders.

The experiments demonstrated a notable potential of the chat model TinyLlama in topical modeling since it showed the capability to identify 14 different topics while achieving clustering scores comparable to those obtained with BERT. Also, the reduced set of BERT-base encodings demonstrated the ability to cluster the publications into 13 topics, with significant enhancement in clustering scores, achieving score improvement of 116%, 135%, and 38% in Silhouette, Calinski-Harabasz, and Davies-Bouldin scores respectively, compared to the conventional BERT encodings.

In summary, this analysis has provided valuable insights into effective methods for topical clustering, highlighting the strengths of large chat models such as TinyLlama, as well as the benefits of dimensionality reduction with autoencoders. These findings contribute to the existing methods and tools available for in-depth exploration of text and topic modeling.

Future analyses may explore the capabilities offered by larger Llama2 models, such as those with 7B, 13B, and 70B parameters, to further enhance the efficacy of topical clustering tasks.

Bibliography

[1] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.

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

[3] P. Zhang, G. Zeng, T. Wang, and W. Lu, "Tinyllama: An open-source small language model," *arXiv preprint arXiv:2401.02385*, 2024.

[4] H. Touvron, T. Lavril, G. Izacard, X. Martinet, M.-A. Lachaux, T. Lacroix, B. Rozière, N. Goyal, E. Hambro, F. Azhar *et al.*, "Llama: Open and efficient foundation language models," *arXiv preprint arXiv:2302.13971*, 2023.

[5] T. Brown, B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell *et al.*, "Language models are few-shot learners," *Advances in neural information processing systems*, vol. 33, pp. 1877–1901, 2020.

[6] M. Lewis, Y. Liu, N. Goyal, M. Ghazvininejad, A. Mohamed, O. Levy, V. Stoyanov, and L. Zettlemoyer, "Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension," *arXiv preprint arXiv:1910.13461*, 2019.

[7] L. Banarescu, C. Bonial, S. Cai, M. Georgescu, K. Griffitt, U. Hermjakob, K. Knight, P. Koehn, M. Palmer, and N. Schneider, "Abstract meaning representation for sembanking," in *Proceedings of the 7th linguistic annotation workshop and interoperability with discourse*, 2013, pp. 178–186.

[8] Z. Bettouche and A. Fischer, "Topical clustering of unlabeled transformer-encoded researcher activity," *Bavarian Journal of Applied Sciences*, no. 6, pp. 504–525, 2023.

[9] M.-H. Weng, S. Wu, and M. Dyer, "Identification and visualization of key topics in scientific publications with transformer-based language models and document clustering methods," *Applied Sciences*, vol. 12, no. 21, p. 11220, 2022.

[10] A. Subakti, H. Murfi, and N. Hariadi, "The performance of bert as data representation of text clustering," *Journal of big Data*, vol. 9, no. 1, pp. 1–21, 2022.

[11] M. Ait-Saada, F. Role, and M. Nadif, "How to leverage a multi-layered transformer language model for text clustering: an ensemble approach," in *Proceedings of the 30th ACM International Conference on Information & Knowledge Management*, 2021, pp. 2837–2841.

Bibliography

[12] J. Gatto and S. M. Preum, "Not enough labeled data? just add semantics: A data-efficient method for inferring online health texts," *arXiv preprint arXiv:2309.09877*, 2023.