TOPIC MODELING OF OPENML DATASET DESCRIPTIONS

Ivan, Germanov
ID: 1483838
Artificial Intelligence and Data Engineering Lab
Eindhoven University of Technology
i.germanov@student.tue.nl

February 29, 2024

ABSTRACT

Provide an informative summary of your proposal (topic, approach and potential importance of the results) in no more than three hundred words. Make sure to provide an informative and relevant abstract, as this is often the first part of your proposal that reviewers will read. The abstract should clearly describe what you are going to investigate, why you are going to investigate this subject and which results you expect to find. The abstract should have no more than 300 words and should contain a single paragraph. This proposal should be self-contained and has a limit of pages per section. **Title page, Abstract, Sections 1 and 2 have a combined limit of 8 pages. Section 3 has a limit of 8 pages. Section 4 has a limit of 2 pages. References and Appendices are unlimited.** You should not change the overall format considerably. Please do not change the margins nor introduce considerable LaTeX code that try to compress the content. You may use the appendices for any further information that you want to provide. While there is no page limit in the appendix, reviewers are not obliged to read all your appendix content.

Wordcount: 190

1 Overall aim and goals

1.1 Motivation and Challenges

Topic modeling is a rapidly growing field with applications in various contexts that include text corpora - social media posts [1, 2, 3], books [4], newspapers [5, 6, 7], legal documents [8, 9], research papers [10] and financial reports [11, 12], to name a few. Topic models can take a large corpus of documents as input and extract the latent topics present therein [13]. A topic refers to a recurring pattern of words or phrases that commonly occur together in a set of documents [14].

Churchill and Singh [15] define a topic model to be a mathematical model that takes as input a set of documents D, and returns a set of topics T that represent the content of D in an accurate and coherent manner. The documents within the collection can subsequently be tagged with these identified topics. This process enables users to discern the importance of each topic both within individual documents and across the entire collection.

OpenML [16] is a platform designed for machine learning researchers to share and manage data. It facilitates global collaboration by allowing users to present new datasets for analysis and share their findings, including code, models, predictions, and evaluations. OpenML ensures the clear definition of tasks and organizes all contributions online for easy accessibility, reuse, and discussion.

For each dataset, OpenML provides a dedicated page that contains detailed information such as a general dataset description, attribution details, and characteristics of the data, as well as statistics on the data distribution. Additionally, OpenML supports the use of tags on datasets, facilitating easier filtering and searchability.

1.1.1 Motivation

In OpenML, datasets are currently categorized using manual tagging. However, many datasets lack semantic tags that are readable by humans. This situation presents an opportunity for a Master's thesis project aimed at developing an automated system for tagging datasets. Given that most datasets come with descriptions, applying topic modeling to extract topics can be an innovative approach to generate and assign relevant tags.

Applying topic modeling to extract topics as tags could enhance how users interact with the OpenML platform. Specifically, it could make the process of searching and filtering through the extensive collection of datasets more efficient, thus enhancing dataset discoverability. The addition of semantic tags based on the topics identified in the descriptions could also lead to better organization and management of datasets, thereby improving data governance on the platform.

Automating the process of tagging can save considerable time for researchers and data scientists who would otherwise have to tag datasets manually. This method ensures consistency in the tags applied and enriches the datasets' metadata, making them more useful and accessible.

Furthermore, previous work by Das has shown the potential of using scripts to automate the tagging of datasets in OpenML [17]. Das's approach involved using dataset descriptions and a predefined list of tags to prompt GPT-3.5-turbo to assign relevant semantic tags to each dataset. This method demonstrated the feasibility of classifying datasets with a set of predefined tags, similar to the dataset tags in the Wolfram Data Repository [18].

This research aims to explore the potential of topic modeling when applied to dataset descriptions, an area that to our knowledge has not been extensively studied. By doing so, it seeks to contribute new insights to the field of topic modeling, which has been applied in various contexts but not extensively in the categorization of datasets based on their descriptions.

1.1.2 Challenges

One challenge encountered is the absence of comprehensive descriptions for datasets on OpenML. A significant number of these datasets lack descriptions or possess only short ones. Furthermore, the available descriptions often present difficulties in topic extraction due to their semi-structured nature, incorporating elements such as author names, dates, and attribute information. To address this issue, it may be necessary to . To address this issue, it may be necessary to identify additional datasets with adequate descriptions for training the topic model, while excluding datasets with inadequate descriptions from topic extraction and tagging process.

The absence of ground truth values for evaluating the performance of the proposed topic model presents another challenge. A ground truth value, such as "Economics" for a dataset containing longitudinal S&P 500 price data, is an example of what is missing from the OpenML datasets. Consequently, identifying and selecting appropriate evaluation metrics to measure the extracted topics and tags' quality is necessary.

1.2 Broad Literature Analysis

1.2.1 Early Foundations and Probabilistic Models

The origins of topic models can be traced back to the early 1990s with the development of Latent Semantic Indexing (LSI), introduced by Deerwester et al. [19]. LSI creates a word-document matrix from a given vocabulary and a collection of documents. This matrix records how frequently each word in the vocabulary appears in the documents. The key step in LSI is the use of singular value decomposition (SVD). SVD compresses the dimensionality of documents, while still maintaining the meaning of the words.

LSI served as a precursor to topic models. In 1999, Hofmann [20] introduced Probabilistic Latent Semantic Indexing (pLSI). In pLSI, the SVD component of LSI was replaced with a generative data model known as an *aspect model*. This change enabled the training of the model using an expectation maximization algorithm. Instead of deriving topics through SVD, Hofmann's approach allowed topics to emerge as probabilistic mixtures of words. These mixtures were based on the joint probabilities of words and documents. This probabilistic framework marked a departure from the earlier matrix factorization approach of LSI and laid the groundwork for more advanced topic modeling techniques.

In 2000, Nigam et al. [21] explored how to integrate unlabeled data into text classification, leading to the development of the Dirichlet Multinomial Mixture (DMM). They employed expectation maximization in conjunction with the Dirichlet distribution [22]. The Dirichlet distribution is a multivariate extension of the Beta distribution. Unlike the Beta distribution, which is defined by parameters α and β, the Dirichlet distribution is characterized by a parameter k. This k represents the number of dimensions in the Dirichlet distribution. These dimensions collectively form a

normalized probability distribution, which is adjusted using the parameter α. The Dirichlet distribution is particularly suitable for topic modeling, as each topic can be represented as one of the k dimensions in the distribution.

1.2.2 Latent Dirichlet Allocation

The term "topic model" was coined by Blei et al. [13] in their seminal 2001 paper on Latent Dirichlet Allocation (LDA). This work adopted the use of a generative model in a similar way to pLSI, but bases its model on the Dirichlet distribution. LDA introduced the concept that documents can be associated with multiple topics, rather than a single topic, as was the case with previous models. Furthermore, a key advancement of LDA was its capacity to be applied to new, unseen documents. The influence of LDA in topic modeling has been substantial, leading to numerous works improving upon LDA, and on the creation of LDA variants adapted for various tasks [23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34].

Most topic models, including LDA, are unsupervised. However, for some tasks, supervision is required. To address this, a variant of LDA named Supervised Latent Dirichlet Allocation (sLDA) was developed, which transforms LDA into a supervised model Mcauliffe and Blei [35]. sLDA extends LDA by associating a response variable with each document. The aim of sLDA is to discover latent topics that are both descriptive of the documents and predictive of the response variable. It achieves this by using a generalized linear model to connect the response variable with the topic proportions in each document.

The evolution of topic models has been influenced by changes in the types of data being analyzed, as noted by Churchill and Singh [15]. In the 2000s, the primary data sources for topic models were scientific articles, books, and newspapers. However, the landscape has shifted in recent years, with an increasing focus on digital and social media content such as tweets, blog posts, and Reddit posts.

Despite the change in data types, LDA and its variants have remained prevalent in the field. These models have been recognized as best practices for various data types and continue to be relevant. However, the field of topic modeling has progressed to include newer models beyond LDA.

1.2.3 Non-negative Matrix Factorization and Other Topic Models

Non-negative Matrix Factorization (NMF) represents one of the newer advancements in topic modeling, as highlighted by Churchill and Singh [15]. NMF is a technique where an original matrix, consisting of non-negative values, is decomposed into two distinct matrices. The fundamental principle of NMF is that the product of these two matrices approximates the original matrix. This decomposition is a form of dimensionality reduction. The large original matrix typically represents a set of documents, with each document being a vector of words. The two resultant matrices in NMF are the topic-word matrix and the topic-document matrix. The topic-word matrix shows the association between topics and words, while the topic-document matrix shows the relationship between topics and individual documents.

The application of NMF in topic modeling was first demonstrated by Shahnaz et al. [36]. They showcased the potential of NMF as a tool for extracting topics from a collection of documents. Following this, [37] expanded the application of NMF to temporal topic models. Yan et al. [38] later augmented NMF by replacing the document-term matrix with a term correlation matrix to detect topics in short texts.

In their 2013 study, Yan et al. [39] presented the Biterm Topic Model (BTM). This model is specifically tailored for analyzing short texts, like tweets and social media updates. Instead of focusing on patterns within entire documents, BTM works by identifying and analyzing pairs of words, termed 'biterms'. These biterms are formed based on a distribution of topics and words.

In 2015, Quan et al. [40] introduced the Self-Aggregating Topic Model (SATM), aimed at improving the topic modeling of short texts. SATM comprises two steps. Initially, it runs LDA on short texts. Subsequently, it uses the topics generated in the first step to create longer pseudo-texts. A pseudo-text in this context refers to the concatenation of shorter documents into a single, longer document.

In 2014, Yin and Wang [41] augmented the Dirichlet Multinomial Mixture (DMM) by introducing the Gibbs Sampling DMM. This adaptation incorporated the Gibbs sampling algorithm, specifically aimed at more effective modeling of short texts. This innovative approach to the established DMM model paved the way for further advanced variations of DMM.

Following this advancement, Li et al. [42] proposed two models: the Generalized Polya Urn Dirichlet Multinomial Mixture (GPUDMM) and the Generalized Polya Urn Poisson-based Dirichlet Multinomial Mixture (GPUPDMM). These models, similar to the approach by Nguyen et al. [43], integrate word embeddings into the classic DMM model. In the GPU approach, when a word is selected, a copy of that word along with similar words are added back to the topic. This mechanism leads to clusters of similar words rising to the forefront of a topic, thus creating topics with

more coherent and related sets of words. Regarding the DMM aspect of their models, Li et al. directly draw from [41]'s GSDMM Yin and Wang.

Building upon DMM, Li et al. [44] developed the Common Semantics Topic Model (CSTM). In this model, they introduced a concept known as 'common topics', designed to capture words that are prevalent across all topics. CSTM then creates topics by combining words from a single specific topic with words from these common topics.

1.2.4 Graph-based Topic Models

Graph-based models are another approach to topic modeling that followed LDA. In these models, words are represented as nodes in a graph, with their co-occurrences indicated by weighted edges. This method diverges from previous generative models by not assuming any underlying topic distribution, which facilitates the discovery of topics of varying sizes.

Cataldi et al. [45] were among the first to implement a graph-based model using a directed graph to detect emerging topics. Subsequently, de Arruda et al. [46] developed a topic model known as Topic Segmentation (TS), which was based on an undirected graph. In 2018, the Topic Flow Model (TFM) was introduced by Churchill et al. [47], applying graph-based methods to track the evolution of topics over time. Following this, Churchill and Singh [48] proposed the Percolation-based Topic Model (PTM), a graph-based model designed to detect topics within noisy datasets.

1.2.5 Word Embedding Topic Models

The integration of natural language processing (NLP) techniques into topic models marks a recent advancement in the field, diverging from earlier statistical models like LDA. A key development in this area has been the use of pre-trained NLP models to augment the capabilities of unsupervised topic models.

According to Almeida and Xexéo [49], the most prominent form of NLP models employed in this context is word embedding spaces. The inception of word embeddings can be traced back to the early 2000s with the work of Bengio et al. [50], who proposed a neural model for learning distributed representations of words.

A seminal study in the field of NLP for creating word embeddings is Word2Vec by Mikolov et al. [51]. This study demonstrated the efficacy of word vectors in identifying semantically similar words. Word2Vec itself comprises two distinct architectures that facilitate the learning of high-quality word embeddings: Skip-gram and Continuous Bag of Words (CBOW).

Nguyen et al. [43] later enhanced the LDA and DMM models by incorporating word embeddings, resulting in two new models: Latent Feature LDA (LF-LDA) and Latent Feature DMM (LF-DMM). In these models, they added a word embedding component to the topic-word distribution of LDA and DMM. LF-LDA and LF-DMM maintain the original structure of LDA and DMM, but integrate a word embedding for each word in their distributions. When generating words for a document, the model can select words either from the topic's distribution or from the word embedding associated with that topic. This approach effectively enlarges the selection pool of words. The addition of the word embedding component improves the models' performance, especially when dealing with short texts.

Qiang et al. [52] developed the Embedding-based Topic Model (ETM), which leverages Word2Vec and introduces a new distance metric known as Word Mover's Distance (WMD). WMD calculates the minimum cumulative distance that words in one document need to travel to match the closest corresponding words in another document. After computing WMD, ETM combines short documents into longer pseudo-texts. Subsequently, LDA is applied to these pseudo-texts to determine topic assignments. The model then constructs an undirected graph to create a Markov Random Field. In this framework, similar words appearing in the same pseudo-text are more likely to be assigned to the same topic.

Bunk and Krestel [53] proposed another enhancement to LDA, termed Word Embedding LDA (WELDA). This approach involves integrating a pretrained word embedding model with a slightly modified version of LDA.

Li et al. [54] further adapted the Dirichlet Multinomial Mixture (DMM) model to better suit short texts, creating the Laplacian DMM (LapDMM). This model integrates variational manifold regularization to maintain the local neighborhood structure inherent in short texts. Before training LapDMM, a graph is constructed to measure the distances between documents, identifying their nearest neighbors. This graph's Laplacian matrix is then utilized as a constraint in the topic assignment process. This ensures that documents assigned to the same topic contain words that are located in similar neighborhoods within the graph. To calculate the distances between documents, the authors employ Word2Vec along with WMD.

In 2016, Miao et al. [55] introduced the Neural Variational Document Model (NVDM), which employs a neural network to perform a multinomial logistic regression. This process is used to generate a word embedding for each document.

In the same year, Moody [56] developed Ida2Vec, a model that integrates Word2Vec with the traditional LDA model. Ida2Vec generates vectors for both documents and words, enabling the measurement of similarity between documents as well as between documents and words or phrases. In this model, each topic is represented as a vector in the same space as the word and document vectors. The resultant topic vector can then be compared with word vectors to identify words most closely related to the topic.

Le and Mikolov later extended their Word2Vec model by introducing Doc2Vec [57]. Doc2Vec extends Word2Vec by introducing a novel framework that allows for the generation of vector representations not just for words, but for larger blocks of text such as sentences, paragraphs, or entire documents. While Word2Vec models (both Skip-gram and Continuous Bag of Words (CBOW)) are efficient at capturing the semantic similarity between words based on their context, they do not directly provide a method for aggregating these word vectors into meaningful representations for larger texts. Doc2Vec addresses this limitation through its architecture, enabling the capture of document-level context.

In 2020, Angelov [58] introduced Top2Vec. Top2Vec extends Word2Vec and Doc2Vec by leveraging their distributed representations of words and documents to model topics. It utilizes Doc2Vec to create semantic embeddings of documents and words, embedding them jointly in the same space, where the proximity between document and word vectors represents semantic similarity. This joint embedding allows for the identification of dense clusters of document vectors, assumed to represent topics. Top2Vec calculates topic vectors as centroids of these clusters and identifies topic words by finding the closest word vectors to each topic vector. This approach provides a more nuanced understanding of topics by exploiting the semantic relationships inherent in the distributed representations of words and documents.

In 2016, Zuo et al. [59] improved their original STM model by introducing the Pseudo-document-based Topic Model (PTM). PTM aggregates multiple short texts into a single pseudo-document. This approach results in a condensed word co-occurrence matrix, which in turn leads to a more accurate approximation of the topics.

In 2017, Bicalho et al. [60] introduced the Distributed representation-based expansion (DREx) technique. This method involves expanding a given document by incorporating the closest n-grams found in the embedding space that are similar to the n-grams present in the document.

Viegas et al. introduced two topic models, CluWords [61] and CluHTM [62], both of which utilize clusters of words and the Term Frequency-Inverse Document Frequency (TF-IDF) method to generate topics. TF-IDF is a widely used metric in text mining that indicates the relevance of a word to a document within a collection or corpus. The core concept in these models is a 'CluWord', which is essentially a cluster of words defined within an embedding space. The process begins by determining CluWords for each word in the vocabulary. Then, in each document, every word is replaced by its corresponding CluWord. Following this replacement, the TF-IDF values of the CluWords are calculated. CluHTM extends the concept of CluWords by combining it with NMF to facilitate hierarchical topic modeling.

Dieng et al. introduced two models: the Embedded Topic Model (ETM) [63] and the Dynamic Embedded Topic Model (D-ETM) [64]. In both models, topics and words are represented within an embedding space. Like LDA, ETM draws a topic for each document, but it diverges from LDA by using the logistic-normal distribution instead of the Dirichlet distribution. For each word in a document, ETM assigns a topic, and then the observed word is drawn based on this topic assignment. This means that words are selected from their embeddings rather than based on their proximity to other words in the document. The D-ETM model extends this concept by adding a time-varying component to the framework. It runs the generative process at each time step, maintaining k topics at each step, but all of these topics are still projected onto the same embedding space.

1.2.6 Transformer-based models

The Transformer model [65], introduced by Vaswani et al., marked another seminal step in the field of NLP. It is an architecture that significantly improves upon the efficiency and effectiveness of previous models for machine translation and other sequence-to-sequence tasks. Groundbreaking for its exclusive use of attention mechanisms, the Transformer eliminates the need for recurrence and convolutions. It obviates the sequential data processing inherent in recurrent neural networks (RNNs) and the fixed receptive fields of convolutional neural networks (CNNs), enabling much greater parallelization of computation. This architecture sets new state-of-the-art benchmarks on translation tasks, demonstrating its superior ability to handle long-range dependencies within text. The Transformer model has since influenced the development of numerous NLP models and frameworks, marking a pivotal shift in the approach to sequence modeling and machine learning. Its relevance to topic modeling is indirect but significant. Transformer models process text in a way that captures deeper semantic meanings, which can be leveraged for identifying coherent topics in large text corpora.

Following Vaswani et al. [65], OpenAI introduced the GPT-1, GPT-2 and GPT-3 models [66], [67], [68]. These models set state-of-the-art results, primarily through increases in the size of the training datasets together with an increased model size (parameter count).

In their 2019 work, Devlin et al. [69] introduced BERT (Bidirectional Encoder Representations from Transformers), a model leveraging the Transformer architecture to pre-train deep bidirectional representations from unlabeled text. Unlike the original Transformer by Vaswani et al. [65], which processes text in a unidirectional manner, BERT enhances understanding by evaluating both left and right contexts across all layers, achieving a more comprehensive grasp of language nuances.

There were a few seminal papers on BERT after Devlin et al.'s paper, namely RoBERTa [70] and Sentence-BERT (SBERT) [71].

In their research, the authors identified that the original BERT model had not been fully optimized in its training regimen. To address this, RoBERTa was developed, refining BERT's training by eliminating the next-sentence prediction objective, utilizing larger mini-batches and higher learning rates, and extending training over a substantially larger dataset. These strategic enhancements markedly boosted RoBERTa's performance, setting new benchmarks across a wide spectrum of NLP tasks.

In the former paper, the authors noticed that the original BERT model was undertrained, and so used RoBERTa refines BERT's training process by removing the next-sentence prediction objective, using larger mini-batches and learning rates, and training on much more data. These optimizations lead to significantly improved performance across a variety of NLP benchmarks.

Sentence-BERT, proposed by Reimers and Gurevych [71], adapts BERT for efficient computation of sentence embeddings. By using a siamese network structure, SBERT generates embeddings that can be compared using cosine similarity, facilitating tasks like semantic textual similarity assessment and clustering with reduced computational resources and time.

In the context of topic modeling, Thompson and Mimno [72] later proposed using BERT [69] to produce topics. The authors use k-means to cluster tokens observed in the data set based on their contextual vectors drawn from BERT. This clustering task differs from previous models such as GloVe [73] and Word2Vec which are context-free embedding spaces with a single embedding representation for each word.

Grootendorst [74] later devised BERTopic, which is a state-of-the-art topic model that is based on the clustering idea. BERTopic employs pre-trained transformer models for creating document embeddings, clusters these embeddings, and utilizes a class-based variation of TF-IDF, termed c-TF-IDF, for generating topic representations. This method ensures the generation of coherent topics and remains competitive in benchmarks against both traditional and modern clustering-based topic modeling approaches.

1.3 Formulation of the problem and objectives

To address the challenge of enhancing OpenML dataset discoverability through automated tagging, this project proposes to leverage topic modeling techniques on dataset descriptions. The aim is to develop a system that automates tag generation while enriching dataset metadata. In addition, automating the process of tagging can save considerable time for researchers and data scientists who would otherwise have to tag datasets manually. Key objectives include analyzing current topic modeling approaches, identifying and training a topic model for topic extraction, and establishing evaluation metrics and baselines for model performance.

Objectives - Based on the importance of training an effective topic model for extracting useful topics for tag generation, we define the following objectives:

1. **OB1** - Perform a comprehensive analysis of the different approaches for extracting coherent tags from the OpenML dataset descriptions.
2. **OB2** - Develop and implement a topic model that effectively identifies and extracts high-quality topics suitable for generating tags.
3. **OB3** - Research, define and implement appropriate evaluation metrics and benchmarks to compare the efficacy of the proposed topic model.

2 Research approach

2.1 Overall methodology and decomposition

The research involves three phases: research, development, and evaluation.

During the research phase, the focus is on understanding the existing literature related to the project. This phase also includes setting up a development environment. This environment will be important for building models, testing them using evaluation metrics, and for defining baseline models. Additionally, this phase involves analyzing potential implementation methods and conducting exploratory data analysis on OpenML dataset descriptions.

The development phase centers on preparing the OpenML dataset descriptions for use. This preparation involves cleaning and preparing the data. In this phase, the insights gained from the research phase are applied to create topic models using the OpenML dataset descriptions.

Finally, in the evaluation phase, appropriate metrics are selected to assess the model's performance. These metrics help in understanding the quality of the developed model. Also, baseline models will be defined to be used as a point of comparison with the proposed model.

At each step, the findings will be documented in the Master's thesis report.

2.2 Methods and techniques

To be more precise about the methods and techniques which will be used during this research:

- **Data cleaning** - We will begin by cleaning the dataset descriptions to remove noise. This may include purging inadequate data points (e.g., descriptions that are too short), removing of stop words, stemming and lemmatization (for models which require them, such as LDA).
- **Evaluation metrics and baselines** - We will define appropriate evaluation metrics and baselines. Evaluation metrics may include coherence, perplexity, and diversity, but manual inspection of the topics will also be used to assess the quality of the generated topics and their alignment with the datasets' content. Baselines will include traditional topic modeling approaches such as LDA and NMF. These models are known for their capability in identifying thematic structures within large text corpora and will serve as benchmarks to evaluate the performance of the proposed model.
- **Topic modeling** - After establishing baselines, we focus on employing BERTopic, since it leverages transformer-based embeddings for improved topic detection. BERTopic is expected to offer superior performance in terms of topic coherence and relevance due to its nuanced understanding of context and semantics.
- **Model optimization** - Both baseline models and BERTopic will undergo hyperparameter tuning to optimize performance.
- **Tag generation** - Relevant and coherent topics identified will be converted into tags.

2.3 Research plan and timeline

The Gantt chart (see Figure 1) shows the scheduled timeline for conducting this research. Each row in the chart corresponds to a particular task within the research, and each column represents one week. The outcome of each phase is classified as either a milestone or a deliverable. Detailed information for each task is provided as follows:

- (**A**) Literature analysis - This involves thorough reading of the relevant literature on topic modeling. **Milestone:** Better understanding of current state-of-the-art topic models, and gaining deeper insight into methods for modeling the OpenML dataset descriptions.
- (**B**) Environment setup - This task involves establishing the coding environment, which includes installing necessary Python packages for topic modeling and configuring the OpenML library. Additionally, a brief exploration of the OpenML Python API will be performed. A version control system (Git) will be set up. **Milestone:** Operational development environment.
- (**C**) Exploratory data analysis (EDA) - In this phase, an EDA will be conducted on the dataset descriptions. This will involve identifying challenges and limitations inherent in the data. A statistical analysis will be performed. **Deliverable:** Explanation, graphs and charts describing the data and its limitations.
- (**D**) Data cleaning and preparation - During this phase, the tasks of data cleaning and data preparation will be performed, making the data suitable for use by the topic models. **Deliverable:** Explanation, graphs and charts of the prepared dataset.
- (**E**) - Experiment design - In this phase, suitable evaluation metrics and baselines will be defined. The proposed model will be evaluated based on those metrics and baselines. **Deliverable:** Definition and justification of experiment design.

- (**F**) - Topic model development - This main phase focuses on creating the topic model that will extract semantic tags from the dataset descriptions. This will include hyperparameter tuning. **Deliverable:** A detailed explanation of the model's steps and architectural decisions, hyperparameters, along with the results when applied to OpenML descriptions.
- (**G**) - Experiment results - during this phase, the developed topic model will be evaluated based on the evaluation metrics. Baseline model(s) will be initialized for comparison with the model. **Deliverable:** Results from the evaluation metrics and comparison between the developed topic model and the baseline model(s).
- (**H**) and (**I**) Defense preparation and Defense presentation - These steps are designated for preparing and presenting the project defense. **Milestone:** Preparation for the Thesis presentation.
- (**J**) Thesis writing - This phase will take place throughout the entire research process. The objective is to document the discoveries, examine the data, summarize the conclusions, identify challenges, and record the outcomes of the research. **Deliverable:** Master's thesis.

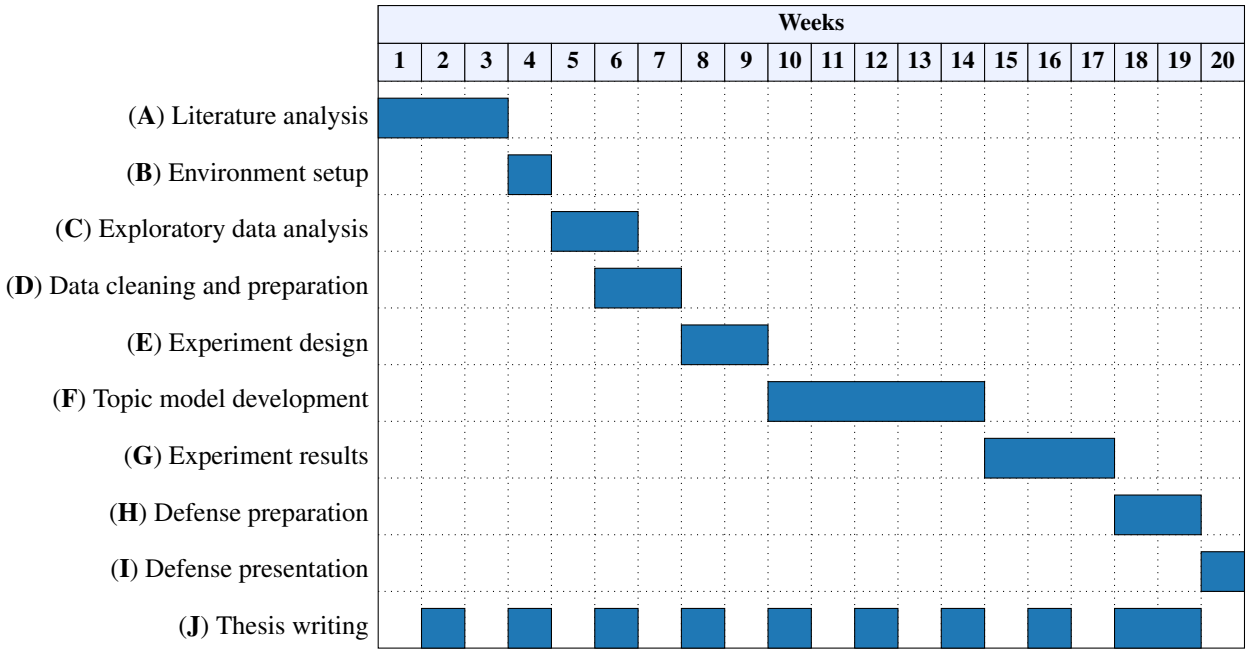

Figure 1: Gantt chart: Research timeline

2.4 Identified risks and their mitigation

Table 1 presents the identified risks for this project. The second column denotes the likelihood of each specific risk occurrence. The third column provides concise descriptions for each identified risk. The last column outlines potential mitigations for each risk.

2.5 Knowledge utilisation/ valorisation / expected contributions and impact

This research is expected to contribute to the evolving field of topic modeling. Existing applications of topic modeling span a range of contexts involving text corpora. Yet, there appears to be a lack of research applying topic modeling to dataset descriptions. Addressing this gap could enhance the originality of the Master's thesis. The study aims to evaluate the efficacy of topic models in the context of dataset descriptions and to identify potential challenges and considerations for future researchers or practitioners undertaking similar work.

In the context of OpenML, this research could improve the user experience by facilitating practitioners' efficient filtering of OpenML's thousands of datasets. The process of filtering could enhance dataset discovery, and the incorporation of semantic tags could improve data governance by introducing a more structured approach to dataset organization within the OpenML platform.

Table 1: Project risks

Risk label	Likelihood	Risk description	Risk mitigation
Low model performance	Medium	The implemented topic modeling approach for OpenML dataset descriptions may perform sub-optimally.	Maintain regular communication with supervisors to discuss project progress and explore alternative solutions as needed.
Low data quality	Medium	The OpenML dataset descriptions may be too short or contain many undesirable artifacts (e.g., author names, arcane symbols, or other uninterpretable by topic models text). This may indirectly lead to low model performance.	Apply dataset cleaning prior to training the topic models (e.g., removal of too short descriptions, identification and removal of artifacts, stemming, lemmatization). Perform an initial exploratory data analysis (EDA) to ascertain the data quality.
Hardware Limitation for Topic Modeling	Medium	The student's personal computer may not possess the requisite computational capacity to run certain topic models, particularly those involving Large Language Models (LLMs).	The student will explore the use of quantized models [75] to reduce computational demands. Should this approach prove insufficient, the student will request access to the hardware facilities provided by the TU/e. Alternatively, a subscription for Google Colab will be purchased, which offers the required computational resources.
Time constraints	Medium	Pertains to the possibility that the student may be unable to fulfill the research requirements within the designated time frame.	The planning outlined in the Gantt chart (Figure 1) will be followed. Should there be deviations from the planned activities that require additional time, a request for an extension of will be submitted.
Unavailability	Low	Arises in the event that the student becomes unavailable, leading to an inability to complete the Thesis project.	The supervisors will undertake the responsibility of identifying and engaging another Master's student to assume the project and drive it to completion or abandon the project.
Ill-defined scope	Low	Associated with the student's challenge in accurately defining the project's scope, resulting in an overly ambitious (too large) or an insufficiently comprehensive (too small) scope.	The student will collaborate with the supervisors to define the project's scope and goals. Additionally, the student will establish a schedule of deliverables and milestones.
Lack of communication	Low	Arises from insufficient communication between the student and the supervisors, leading to misunderstandings or gaps in the supervision process.	Bi-weekly meetings will be organized to discuss project progress and outline activities planned for the forthcoming weeks. Additionally, frequent contact will be maintained through MS Teams/Slack to facilitate real-time communication and prompt resolution of any emerging issues.

Furthermore, the automation of semantic tagging could offer considerable time savings for researchers and data scientists, who would otherwise exert effort on manual tagging. This approach ensures the use of consistent tags and enriches the metadata, further advancing the utility and accessibility of datasets.

3 Evidence that your research can succeed

3.1 Background and In-depth Literature Analysis

This section will delve more into the details for the models, benchmarks and evaluation metrics which will be used during the course of the Master's thesis.

Namely, the plan is to use BERTopic as the base, proposed model, and LDA, NMF and Top2Vec as baseline models. Furthermore, the models will be compared based on the evaluation metrics that are explained in this section.

3.1.1 LDA

[13]

3.1.2 NMF

[36], [37], [38]

3.1.3 Top2Vec

A limitation of LDA and NMF is that they disregard semantic relationships between words, thus neglecting context. As a result, text embedding techniques have become popular as an NLP technique.

[58]

3.1.4 BERTopic

Top2Vec simplifies the process of generating topics by clustering embeddings of words and documents. BERTopic [74] is a state-of-the-art topic model that builds on top of the clutering embeddings approach. It employs a variation of c-TF-IDF for classes to generate representations of topics.

BERTopic generates representations of topics through a six-step process. Initially, it transforms each document into an embedding using a pre-trained language model. Before the clustering process, the dimensionality of these embeddings is reduced. Following this, the embeddings are clustered. Subsequently, a bag-of-words representation is generated for each cluster, containing the frequency of every word. Next, topic representations are derived from these clusters using a specialized class-based version of TF-IDF. The final step optionally fine-tunes these topic representations.

While these steps are the default, BERTopic offers a degree of modularity. Each step in the process is relatively independent from the others. For instance, the tokenization step does not depend on the specific embedding model used for document conversion, which provides flexibility in how tokenization is executed.

This flexibility is particularly important during the clustering step. Clustering models such as HDBSCAN are built on the premise that clusters can vary in shape and form. Consequently, employing a centroid-based technique for modeling topic representations may not be appropriate, as the centroid may not accurately reflect the nature of these clusters. In contrast, a bag-of-words approach assumes minimal knowledge about the cluster's shape and form.

As a result, BERTopic is highly modular, maintaining its ability to generate topics across different sub-models. This means that BERTopic effectively allows for the construction of customized topic models. Figure 2 illustrates the six steps of BERTopic, presented from bottom to top. It highlights the possibility of employing various techniques at each step of the process. For example, one could choose between SBERT or SpaCy for document embedding, UMAP or PCA for dimensionality reduction, and GPT or KeyBERT for the fine-tuning phase.

Document embeddings

In BERTopic, documents are transformed into embeddings to create vector space representations for semantic comparison. It is based on the idea that documents sharing the same topic will have similar semantics. For this embedding step, BERTopic utilizes the SBERT framework Reimers and Gurevych [71]. SBERT enables the conversion of sentences and paragraphs into dense vector representations by employing pre-trained language models. This achieves top performance on several sentence embedding tasks [77]. The embeddings are mainly used for clustering documents with semantic similarities rather than directly for topic generation. BERTopic can use any embedding technique, provided the language model used for generating document embeddings is fine-tuned for semantic similarity. Hence, the quality of BERTopic's clustering improves as more advanced language models are developed, allowing BERTopic to evolve alongside advancements in embedding techniques.

Dimensionality reduction

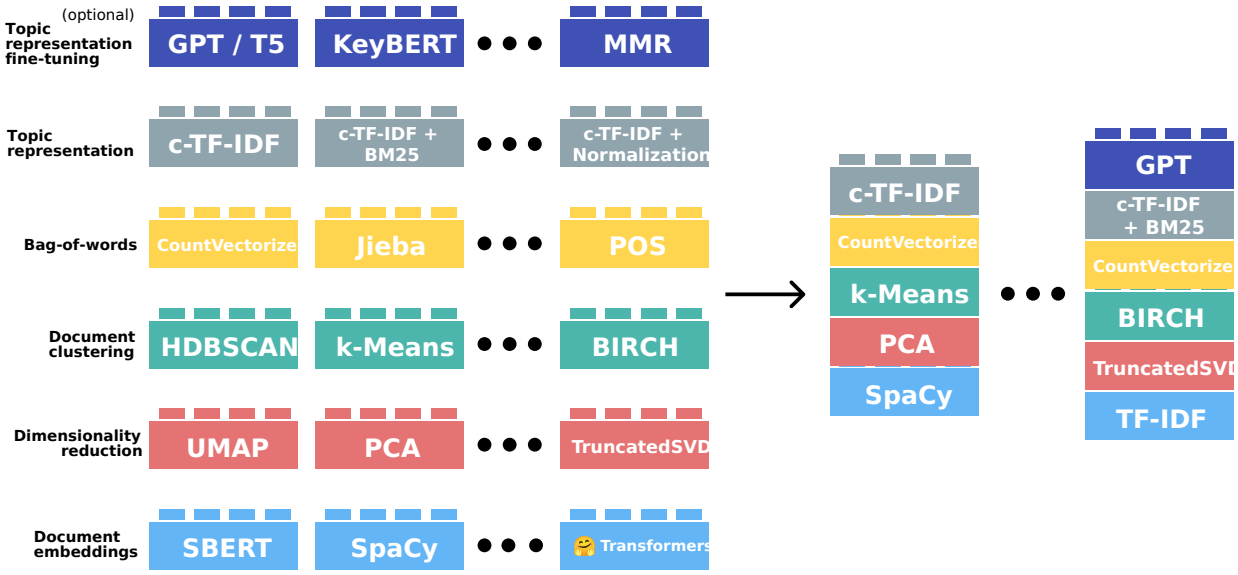

Figure 2: BERTopic modularity (inspired by [76])

As the dimensionality of data increases, the distance to the nearest data point tends to become similar to the distance to the farthest data point [78, 79]. This phenomenon implies that in high-dimensional spaces, the notion of spatial locality becomes unclear, and distances between points show minimal variation. While several clustering methods have been developed to address the curse of dimensionality [80, 81], a simpler strategy involves reducing the dimensionality of embeddings. Although PCA and t-SNE are popular dimensionality reduction techniques, UMAP has been found to better preserve the local and global characteristics of high-dimensional data in its lower-dimensional representations [82]. Furthermore, UMAP's flexibility regarding the dimensions of embeddings allows its application across various language models.

Document clustering

The reduced embeddings are clustered using HDBSCAN [83]. HDBSCAN is built on top of DBSCAN and is designed to identify clusters of various densities by transforming DBSCAN into a hierarchical clustering algorithm. It employs a soft-clustering approach, which allows for the treatment of noise as outliers. This method helps to prevent unrelated documents from being grouped into any cluster, which is expected to improve the quality of topic representations. Furthermore, Allaoui et al. [84] showed that the performance of well-known clustering algorithms, including k-Means and HDBSCAN, can be significantly improved by reducing the dimensionality of high-dimensional embeddings with UMAP, in terms of both clustering accuracy and computational time.

Bag-of-words

Before creating topic representations in BERTopic, it is necessary to select a technique that supports the algorithm's modular nature. When using HDBSCAN, we assume that clusters may vary in density and shape, indicating that techniques based on centroid models may not be suitable. The desired method should ideally make minimal assumptions about the cluster structures.

The process begins by combining all documents within a cluster into a single document, which then represents the entire cluster. Subsequently, the frequency of each word within this single document is counted, resulting in a bag-of-words representation that reflects the word frequencies at the cluster level rather than the individual document level. The adoption of a bag-of-words approach ensures that no assumptions are made about the density and shape of the clusters. Additionally, this representation is L1-normalized to account for the varying sizes of clusters.

Topic representation

The classic TF-IDF [85] method combines term frequency and inverse document frequency to calculate a weight $W_{t,d}$ for term t in document d as follows:

$$W_{t,d} = tf_{t,d} \cdot \log\left(\frac{N}{df_t}\right)$$

Here, term frequency $tf_{t,d}$ represents the frequency of term t in document d, and inverse document frequency measures t's importance across documents, calculated by the logarithm of the ratio of the total number of documents N to the number of documents containing t.

BERTopic extends the TF-IDF concept to clusters of documents by introducing c-TF-IDF. In this approach, documents within a cluster are concatenated into a single document, and the TF-IDF formula is modified for cluster-level representation:

$$W_{t,c} = tf_{t,c} \cdot \log\left(1 + \frac{A}{tf_t}\right)$$

In this formula, term frequency $tf_{t,c}$ now models the frequency of term t within a cluster c, treated as a single document. The inverse document frequency is substituted with an inverse class frequency, which assesses the term's importance to a cluster. This is calculated by the logarithm of the average number of words per cluster A divided by the term's frequency tf_t across all clusters, with 1 added inside the logarithm to ensure positive values. This adaptation of TF-IDF to clusters allows us to model the importance of words in clusters instead of individual documents. Furthermore, by iteratively merging c-TF-IDF representations of less prevalent topics with their closest topics, the total number of topics can be reduced to meet a predefined threshold.

(Optional) Topic representation fine-tuning

After generating the c-TF-IDF representations, we obtain a collection of words that describe a collection of documents. c-TF-IDF is a method for quickly producing accurate topic representations. Nonetheless, the field of NLP is rapidly advancing, with frequent new developments. To make use of these developments, BERTopic offers the option to refine c-TF-IDF topics further using GPT, KeyBERT [86], spaCy [87], and other techniques, many of which are integrated within the BERTopic library.

In particular, the topics generated through c-TF-IDF can be viewed as canditate topics, comprising a set of keywords and representative documents. These can serve as a foundation for further refinement of topic representations. The availability of representative documents for each topic can be useful, as it enables fine-tuning on a reduced number of documents, thereby reducing computational demands. This makes the use of large models like GPT more viable in production environments, often resulting in shorter processing times compared to the steps of dimensionality reduction and clustering.

3.1.5 Evaluation metrics

According to Abdelrazek et al. [14], since topic models can be applied in various application domains, they can be evaluated extrinsically according to how they perform in the domain where they were applied. They can also be evaluated intrinsically by considering the generated topics themselves. Intrinsic evaluation is independent of any specific domain and is thus more general. The different models differ in simplicity, computation efficiency, and modeling assumptions. They accordingly differ in how they perform on different corpora and different applications. There is little consensus on the aspects of topic model evaluation. There have also been different methods to evaluate a specific aspect.

Abdelrazek et al. [14] discuss several evaluation criteria - namely, quality, interpretability, stability, diversity, efficiency and flexibility (Figure 3). Here, we will focus on quality, interpretability and diversity, as they are most appropriate for our use case.

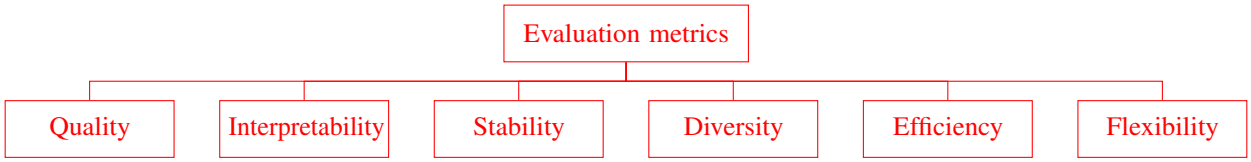

Figure 3: Topic models evaluation criteria [14]

Quality and Perplexity

Perplexity measures the model's capability to generate the documents in corpus based on the learned topics. It measures the model's predictive power but not the latent structure, so perplexity only shows how well the model explains the data. A model is perplexed if the information gain of learning the random variable's outcome is small, so the lower the perplexity, the better the model in explaining the observed documents.

There are some disadvantages to using perplexity as an evaluation metric in this Master's thesis. First, perplexity must be normalized to the vocabulary size in the corpus since it changes with different corpus and topic sizes, whereas BERTopic will not always extract the same number of topics, unless explictly told to reduce the number of topics. Also, perplexity has not been found to be correlated with human judgment [88]. Moreover, non-generative models, such as NMF, will not have a defined perplexity score, since they cannot provide probabilities for word sequences.

Interpretability and Topic coherence

A topic is a discrete distribution over words. This set of words is evaluated for being human interpretable. For this to occur, the generated words should be associated with a single semantic meaning. Topic coherence metrics compute how the top-k words of a topic are related to each other.

Newman et al. [89] assess coherence by observing the lexical similarity between pairs of words. The authors experimented with different similarity measures and found mutual information to be the most consistent performer. The pointwise mutual information, PMI, between word pair (w_i, w_j) is calculated as below:

$$PMI(w_i, w_j) = \frac{\log p(w_i, w_j)}{p(w_i)p(w_j)}$$

It quantifies the discrepancy between the probability of w_i and w_j co-occurring versus the probabilities of their individual occurrences, assuming independence. $p(w_i, w_j)$ is the joint probability that both words w_i and w_j occur together. $p(w_i)$ and $p(w_j)$ are the individual probabilities of the words w_i and w_j occurring in the corpus.

Coherence usually has a trade-off with perplexity [88], optimizing perplexity usually leads to decreased coherence.

Topic diversity

This metric describes how semantically diverse the obtained topics are. One way to define diversity is introduced in [63]. It defines topic diversity as the percentage of unique words in the top 25 words of all topics. So, in general, diversity metrics that measure how diverse the top-k words of a topic are to each other. A topic model should generate diverse topics and score high on this metric. A low score indicates redundant topics, meaning the model could not sufficiently disentangle the corpus's themes. Note that selecting a number of topics in the model affects the topic diversity. A too large number could result in similar topics with overlapping top words. A too small number could result in broad topics with poor interpretability.

Classification evaluation metrics

It is important to note that the evaluation metrics above concern topic modeling as an unsupervised task. If after experimentation we find that our proposed BERTopic model does not perform sufficiently well, we may turn the problem into a (semi-)supervised problem. In this case, different metrics will apply, the most popular of which (besides accuracy, precision, recall and F1 score) are coverage and purity.

Coverage refers to how well the concepts in the document collection are represented. Coverage can be divided into two types of coverage, topic coverage and document coverage. Topic coverage measures assess how representative the topics themselves are, i.e. are the topics in a document collection identified by the model. The most prevalent topic coverage measure is topic recall. Topic recall is the fraction of ground truth topics recovered by the topic model. Document coverage measures evaluate how well documents are represented by topics. Topic model accuracy is a typical measure for evaluating document coverage. It is defined as the fraction of documents that are accurately labeled by the topic model. In order to evaluate topic recall and accuracy, ground truth topics must be available. Topic recall is used more sparingly than topic accuracy.

When a full set of ground truth topics are not readily available, other metrics are used. Purity is the accuracy of the model if documents are always assigned the dominant topic. This metric attempts to penalize models that assign a large number of low probability topics to documents, as opposed to a model that assigns a high probability to a single topic across the document collection.

OCTIS

OCTIS (Optimizing and Comparing Topic models Is Simple) [90] is a unified and open-source evaluation framework for training, analyzing, and comparing topic models over several datasets and evaluation metrics.

It allows the user to optimize the models' hyper-parameters for a fair experimental comparison. The proposed framework provides a topic modeling pipeline (Figure 4). The main functionalities of OCTIS are related to dataset pre-processing, training topic models, estimating evaluation metrics, hyperparameter optimization, and interactive web dashboard visualization.

OCTIS provides several evaluation metrics for topic models, including coherence, significance, diversity and classification metrics.

The optimal hyper-parameter configuration is determined according to a Bayesian Optimization (BO) strategy [91, 92, 93], where the objective can be any of the provided evaluation metrics. Since the performance estimated by the evaluation metrics can be affected by noise, the objective function is computed as the median of a given number of model runs (i.e., topic models run with the same hyperparameter configuration) computed for the selected evaluation metric.

BO is a sequential model-based optimization strategy for expensive and noisy black-box functions (e.g. topic models). The basic idea consists of using all the model's configurations evaluated so far to approximate the value of the performance metric and then selects a new promising configuration to evaluate. The approximation is provided by a probabilistic *surrogate model*, which describes the prior belief over the objective function using the observed configurations. The next configuration to evaluate is selected through the optimization of an *acquisition function*, which leverages the uncertainty in the posterior to guide the exploration.

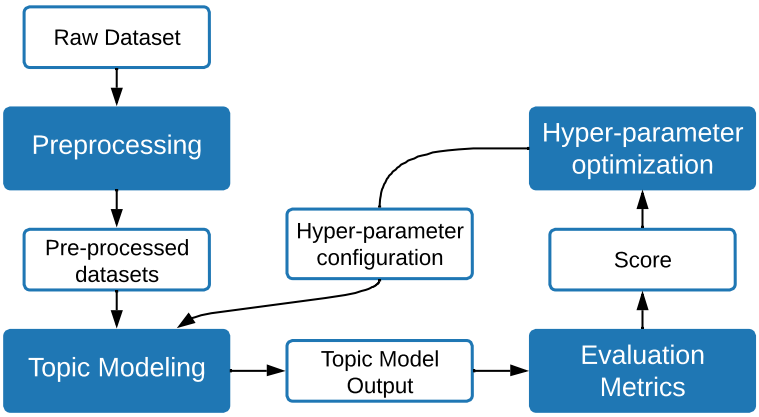

Figure 4: Workflow of the OCTIS framework [90]

OCTIS could prove to be a valuable tool for the Master's thesis, as it provides a unified framework to train the proposed BERTopic model and the baseline models, and to compare them using a variety of evaluation metrics. In fact, in the original BERTopic paper, Grootendorst [74] used OCTIS to evaluate the model's performance.

- Go deeper into the models that will be used. I.e., BERTopic.
- Discuss metrics more in depth.
- Discuss baselines - LDA, T2V, NMF. What could be some other baselines?
Taniya: - For baselines, we can refer to the papers and include models which are generally used as a baseline (for easy comparison)> so LDA, T2V, NMF, etc.

3.2 Preliminary studies and analyses

Two initial developments illustrate the potential of the Master's thesis. The first approach, by Das [17], employs OpenML's dataset descriptions along with GPT-3.5-turbo for dataset tagging. The second development utilized BERTopic, primarily with default parameters.

3.2.1 Tag assignment using GPT 3.5

Previous research by Das has explored the automation of dataset tagging on OpenML using GPT-3.5-turbo for assigning semantic tags based on dataset descriptions and a set list of tags. Das's approach demonstrated the feasibility of classifying datasets with a set of predefined tags, similar to the dataset tags in the Wolfram Data Repository [18].

Specifically, the predefined tags were - *Agriculture*, *Astronomy*, *Chemistry*, *Computational Universe*, *Computer Systems*, *Culture*, *Demographics*, *Earth Science*, *Economics*, *Education*, *Geography*, *Government*, *Health*, *History*, *Human Activities*, *Images*, *Language*, *Life Science*, *Machine Learning*, *Manufacturing*, *Mathematics*, *Medicine*, *Meteorology*, *Physical Sciences*, *Politics*, *Social Media*, *Sociology*, *Statistics*, *Text & Literature*, *Transportation*

To illustrate the potential of using language models for automated dataset classification, the researcher developed a Python script that utilized the OpenAI GPT-3.5-turbo API and spaCy's natural language processing library. The script downloaded the descriptions of all datasets available on OpenML and used the GPT-3.5-turbo API to generate semantic tags for each dataset based on its description and the predefined list of tags.

To process the generated tags, the script employed spaCy to clean and standardize the tags, ensuring that they matched the predefined list of tags. While the script did not include an evaluation of the automated tagging system, it demonstrated the feasibility of using language models for dataset classification and the potential for improving the efficiency of the tagging process.

Table 2 demonstrates the distribution of semantic tags across the OpenML datasets, where "Machine Learning" is the most prevalent tag with a percentage of 20.72%, followed by "Life Science" at 16.22%, and "Chemistry" at 12.87%. These tags indicate the primary areas of focus within the dataset, highlighting the significant emphasis on machine learning techniques, life science research, and chemical studies.

Table 2: Percentage of occurrence for each tag

Tag	**Percentage (%)**
Manufacturing	1.45
Machine Learning	20.72
Mathematics	2.69
Economics	5.01
Education	0.93
Medicine	2.88
Images	1.99
Health	2.35
Demographics	2.77
Life Science	16.22
Agriculture	1.05
Statistics	5.26
Human Activities	0.41
Physical Sciences	0.72
Chemistry	12.87
Text & Literature	0.55

Tag	**Percentage (%)**
Transportation	2.35
Government	1.27
Politics	0.19
No description	2.55
Computer Systems	7.62
Astronomy	0.67
Earth Science	1.12
Social Media	2.07
Meteorology	1.40
Geography	0.99
Language	0.46
Computational Universe	0.90
History	0.12
Culture	0.17
Sociology	0.22

3.2.2 Topic modeling proof of concept using BERTopic

The second stage involved utilizing BERTopic to assess the feasibility of topic extraction. BERTopic was applied in an unsupervised manner to identify latent topics within the dataset descriptions.

Initial cleaning of the datasets involved removing those without descriptions, those with descriptions shorter than 100 characters, and those with repeated dataset descriptions. Datasets with identical descriptions typically represented different versions of the same dataset, where the descriptions did not vary between versions. It should be highlighted that while some datasets do update their descriptions across versions, in the majority of cases, descriptions were very similar.

Figure 5 illustrates the distribution of description lengths by character count. It reveals that most datasets have descriptions under 2000 characters. This observation is not seen as a limitation, given the successful application of topic models on brief texts, such as tweets, in previous studies.

The process then involved transforming descriptions into a word embedding space, using the *all-mpnet-base-v2* [94] sentence transformer from the Sentence-BERT repository [71] *sentence-transformers* as the embedding model. *all-mpnet-base-v2* is an all-round model tuned for many use-cases, trained on a large and diverse dataset of over 1 billion training pairs. The model maps sentences and paragraphs into a 768-dimensional vector space, making it suitable

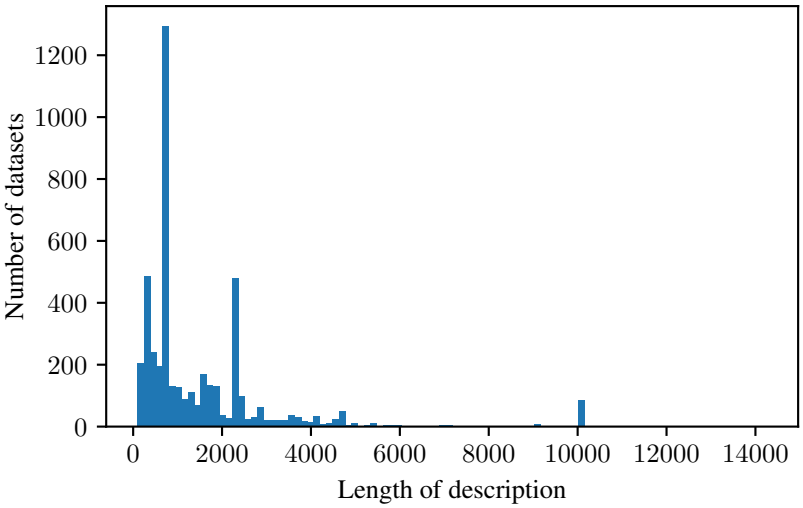

Figure 5: Distribution of dataset description lengths (in characters)

for tasks like clustering or semantic search. It builds upon the pre-trained *microsoft/mpnet-base* [95] model, further fine-tuned using a dataset of 1 billion sentence pairs.

The model was selected due to its top performance in the Performance Sentence Embeddings category, which evaluates models across 14 datasets from the *sentence-transformers* library. It achieved the highest average performance in encoding sentences over 14 diverse tasks from different domains. Additionally, it ranked second in Performance Semantic Search across 6 datasets, a category measuring performance in semantic search tasks. These tasks involve the encoding of queries or questions and paragraphs, accommodating up to 512 word pieces, across a range of subjects.

There is potential for further experimentation with other models available in the *sentence-transformers* library. Given that these models are pretrained, they are expected to perform well out of the box. Additionally, exploring embedding models from other sources, such as Hugging Face Transformers, Flair, spaCy, Gensim, OpenAI, and custom embedding models [96], could potentially give better results.

Following that, the default UMAP model was used for dimensionality reduction, being a good default for this purpose. The specific UMAP parameters used were: `n_neighbors=15`, `n_components=5`, `min_dist=0.0`, and `metric='cosine'`. UMAP is the default choice in BERTopic because it effectively captures both local and global structures of high-dimensional data in lower-dimensional spaces. While UMAP is a sensible default, experimenting with other dimensionality reduction methods, such as PCA or Truncated SVD, may also be viable approaches.

To cluster the reduced-dimensionality embeddings into groups representing similar topics, HDBSCAN was employed. HDBSCAN is well-suited for identifying structures of varying densities. For clustering, we configured HDBSCAN with parameters: `min_cluster_size=10`, `metric='euclidean'`, `cluster_selection_method='eom'`.

For the vectorizer model, we utilized the `CountVectorizer` model, setting `ngram_range=(1,2)`. This configuration allows the model to consider both unigrams, which are single words, and bigrams, which are pairs of consecutive words, to generate a bag-of-words representation.

For the next step, we apply the default c-TF-IDF method to distinguish between the clusters.

Following the generation of c-TF-IDF representations, there is an optional step to fine-tune the c-TF-IDF topics. For this, KeyBERT was used, utilizing cosine similarity to identify sub-phrases within a document that most closely resemble the document as a whole. This allows for the identification of words that most accurately describe the entire document.

The experiment's results reveal that the extracted topics contain descriptive and relevant terms. As illustrated in Figure 6, BERTopic extracted several topics from the descriptions. For example, *Topic 1* encompasses terms associated with economics, particularly financial trading, including *forex*, *candlestick*, *bid_low*, and *bid_high*. *Topic 6* focuses on housing, featuring terms like *residential*, *property*, and *building*. Other topics cover subjects such as cars, twitter data, and transportation.

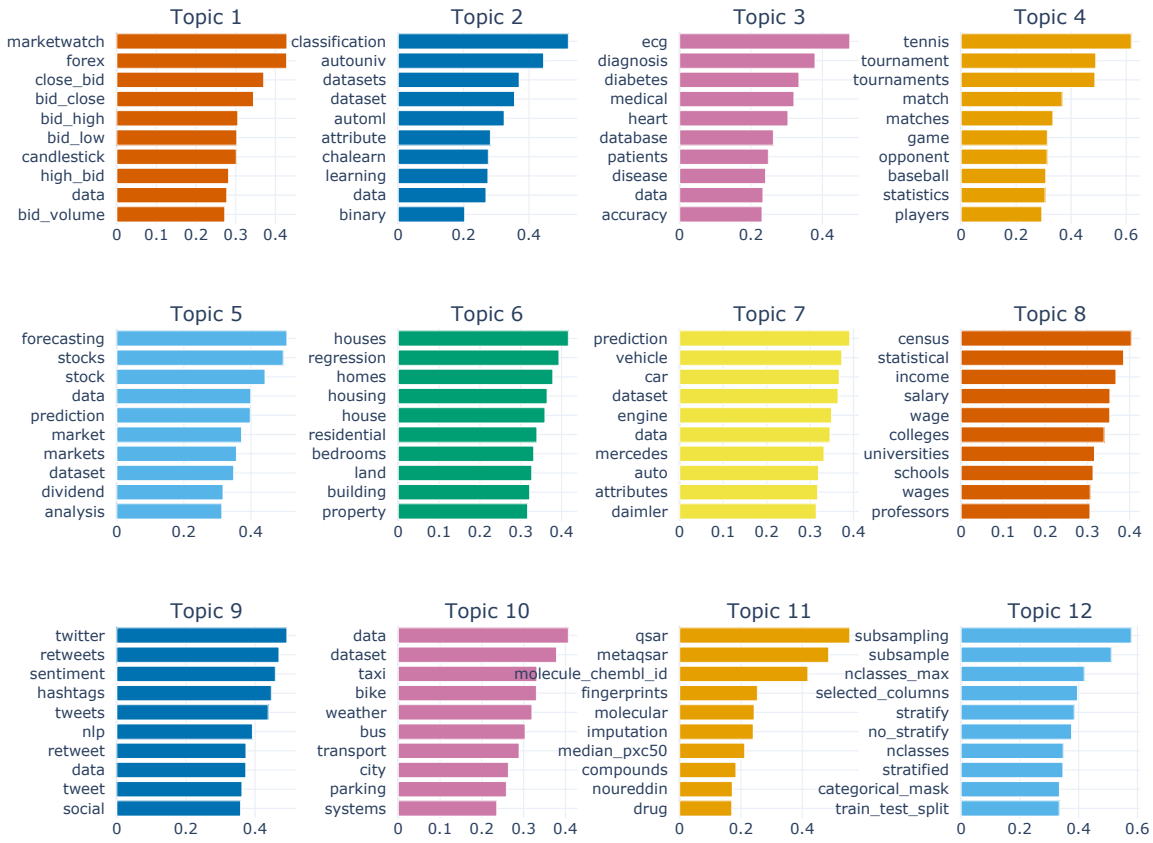

Figure 6: Topic word scores

However, it can be seen that certain topics contain terms that are not particularly useful. These terms often originate from documents that explain attribute values. For instance, *Topic 11* and *Topic 12* feature terms like *median_pxc50*, *stratify*, *no_stratify*, and *train_test_split*.

Additionally, many topics contain terms such as *data*, *dataset* and other similar terms that are not particularly descriptive.

Further experimentation with the model is necessary to address these cases.

4 Other Information

4.1 Data management

This project will leverage existing datasets from the OpenML platform, utilizing its status as an open-source library with datasets that are freely and publicly accessible. Additionally, datasets from other open-source repositories, such as the Wolfram Data Repository [18] or Huggingface [97], may also be employed.

The project will involve the collection and generation of data suitable for reuse. This data is not anticipated to include sensitive information, thus obviating the need for specialized treatment or storage.

Data produced by this project will be made publicly accessible via an open-source repository on a version control platform like GitHub [98].

Regarding the data to be generated, it will contain a pipeline, including the initialization of data fetching (documents), data cleaning for the topic models, the implementation of the proposed topic model along with benchmark models, and the final output, which consists of a set of topic labels pertaining to the documents. Additionally, benchmarks will be generated based on the evaluation metrics.

Figure 7 shows a flowchart of the pipeline, which contains the following sequential steps:

1. **Data Fetching**: This is the first stage where dataset descriptions are downloaded from OpenML (or other sources).
2. **Data Cleaning**: After fetching the data, the next step involves cleaning it. This includes removing noise, correcting errors, and standardizing the format to prepare it for analysis. Data cleaning ensures that the input to the topic model is of high quality, which is crucial for the success of the subsequent modeling steps.

   the next step involves purging inadequate data points, such as excessively short descriptions and duplicates. Stop words are removed for models that require it (LDA). Additionally, the process includes stemming and lemmatization to normalize words to their base forms.
3. **Topic Model**: In this step, the proposed topic model is applied to the cleaned data. The topic model is an algorithm that discovers the underlying thematic structure in the collection of documents. In this case, it will be BERTopic.
4. **Benchmark Models**: Concurrently with the proposed topic model, benchmark models are run. These models represent established or baseline approaches to topic modeling against which the performance of the proposed topic model is compared. This will involve baseline models such as LDA, NMF and Top2Vec.
5. **Topic Labels**: The output from both the topic model and the benchmark models are sets of topics, represented by a cluster of words that are characteristic of a particular topic.
6. **Evaluation**: Finally, the performances of the proposed topic model and benchmark models are evaluated. This can include comparing the topic coherence and diversity, as well as the relevance and interpretability of the topics generated. Evaluation metrics may also include quantitative measures such as perplexity, or qualitative assessments through human judgement.

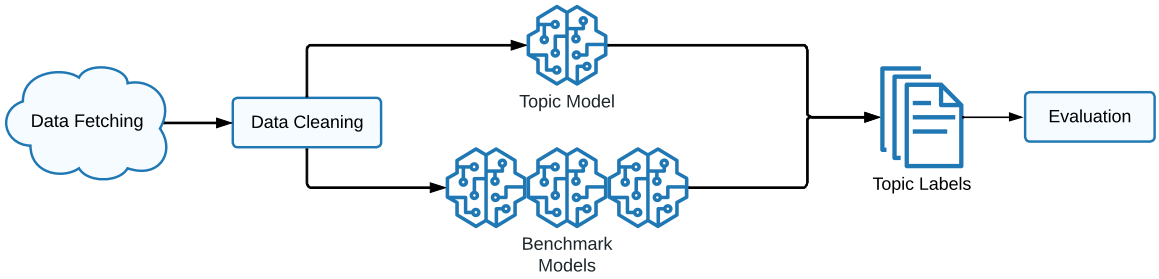

Figure 7: Data pipeline

4.2 Motivation for choice of research group / supervisor / company

In the course of my Master's program, I enrolled in a course titled "Machine Learning Engineering." The course was led by Professor Joaquin Vanschoren, who participates in the Data Mining cluster at the TU/e. Furthermore, I followed several other courses related to machine learning, including "Research Topics in Data Mining," "Machine Learning for Industry," "Data Modeling and Databases," and "Deep Learning." These courses significantly improved my understanding and heightened my interest in the domain of machine learning. By focusing on topic modeling in my Master's thesis, I intend to continue broadening my knowledge and expertise in the field.

Additionally, the open-source status of OpenML played a role in attracting my interest. The prospect of contributing to an open-source project is appealing, since it is beneficial for the developer, practitioner, and research communities. This form of contribution furthers the knowledge in the field and makes it accessible to all.

References

[1] Stephan A. Curiskis, Barry Drake, Thomas R. Osborn, and Paul J. Kennedy. An evaluation of document clustering and topic modelling in two online social networks: Twitter and Reddit. *Information Processing & Management*, 57(2):102034, March 2020. ISSN 0306-4573. doi: 10.1016/j.ipm.2019.04.002.

[2] Michael J. Paul and Mark Dredze. Discovering Health Topics in Social Media Using Topic Models. *PLOS ONE*, 9(8):e103408, August 2014. ISSN 1932-6203. doi: 10.1371/journal.pone.0103408.

[3] Marco Pennacchiotti and Siva Gurumurthy. Investigating topic models for social media user recommendation. In *Proceedings of the 20th International Conference Companion on World Wide Web*, WWW '11, pages 101–102, New York, NY, USA, March 2011. Association for Computing Machinery. ISBN 978-1-4503-0637-9. doi: 10.1145/1963192.1963244.

[4] Krishna Raj P M and Jagadeesh Sai D. Sentiment analysis, opinion mining and topic modelling of epics and novels using machine learning techniques. *Materials Today: Proceedings*, 51:576–584, January 2022. ISSN 2214-7853. doi: 10.1016/j.matpr.2021.06.001.

[5] Carina Jacobi, Wouter van Atteveldt, and Kasper Welbers. Quantitative analysis of large amounts of journalistic texts using topic modelling. In *Rethinking Research Methods in an Age of Digital Journalism*. Routledge, 2018. ISBN 978-1-315-11504-7.

[6] Quintus Van Galen Nicholson, Bob. In Search of America: Topic modelling nineteenth-century newspaper archives. In *Journalism History and Digital Archives*. Routledge, 2020. ISBN 978-1-00-309884-3.

[7] Jani Marjanen, Elaine Zosa, Simon Hengchen, Lidia Pivovarova, and Mikko Tolonen. Topic modelling discourse dynamics in historical newspapers, November 2020.

[8] Raquel Silveira, Carlos G O Fernandes, João A Monteiro Neto, Vasco Furtado, and Ernesto Pimentel Filho. Topic Modelling of Legal Documents via LEGAL-BERT.

[9] James O'Neill, Cecile Robin, Leona O'Brien, and Paul Buitelaar. An analysis of topic modelling for legislative texts. 2016. ISSN 1613-0073.

[10] Claus Boye Asmussen and Charles Møller. Smart literature review: A practical topic modelling approach to exploratory literature review. *Journal of Big Data*, 6(1):93, October 2019. ISSN 2196-1115. doi: 10.1186/s40537-019-0255-7.

[11] Karim El Mokhtari, Mucahit Cevik, and Ayşe Başar. Using Topic Modelling to Improve Prediction of Financial Report Commentary Classes. In Cyril Goutte and Xiaodan Zhu, editors, *Advances in Artificial Intelligence*, Lecture Notes in Computer Science, pages 201–207, Cham, 2020. Springer International Publishing. ISBN 978-3-030-47358-7. doi: 10.1007/978-3-030-47358-7_19.

[12] Silvia García-Méndez, Francisco de Arriba-Pérez, Ana Barros-Vila, Francisco J. González-Castaño, and Enrique Costa-Montenegro. Automatic detection of relevant information, predictions and forecasts in financial news through topic modelling with Latent Dirichlet Allocation. *Applied Intelligence*, 53(16):19610–19628, August 2023. ISSN 1573-7497. doi: 10.1007/s10489-023-04452-4.

[13] David Blei, Andrew Ng, and Michael Jordan. Latent Dirichlet Allocation. In *Advances in Neural Information Processing Systems*, volume 14. MIT Press, 2001.

[14] Aly Abdelrazek, Yomna Eid, Eman Gawish, Walaa Medhat, and Ahmed Hassan Yousef. Topic modeling algorithms and applications: A survey. *Information Systems*, 112:102131, October 2022. doi: 10.1016/j.is.2022.102131.

[15] Rob Churchill and Lisa Singh. The Evolution of Topic Modeling. *ACM Computing Surveys*, 54(10s):1–35, January 2022. ISSN 0360-0300, 1557-7341. doi: 10.1145/3507900.

[16] Joaquin Vanschoren, Jan N. van Rijn, Bernd Bischl, and Luis Torgo. OpenML: Networked science in machine learning. *ACM SIGKDD Explorations Newsletter*, 15(2):49–60, June 2014. ISSN 1931-0145, 1931-0153. doi: 10.1145/2641190.2641198.

[17] Taniya Das. Openml/scripts. https://github.com/openml/scripts/tree/main.

[18] Wolfram Data Repository: Computable Access to Curated Data. https://datarepository.wolframcloud.com/, .

[19] Scott Deerwester, Susan T Dumais, George W Furnas, LANDAUEŔT Landauer, and Richard Harshman. Indexing by latent semantic analysis. *Indexing by latent semantic analysis*, 41(6):391–407, 1990. ISSN 0002-8231.

[20] Thomas Hofmann. Probabilistic latent semantic indexing. In *Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 50–57, Berkeley California USA, August 1999. ACM. ISBN 978-1-58113-096-6. doi: 10.1145/312624.312649.

[21] Kamal Nigam, Andrew Kachites Mccallum, Sebastian Thrun, and Tom Mitchell. Text Classification from Labeled and Unlabeled Documents using EM. *Machine Learning*, 39(2):103–134, May 2000. ISSN 1573-0565. doi: 10.1023/A:1007692713085.

[22] Pierre Simon Laplace (marquís de). *Théorie analytique des probabilités*. Courcier, 1814.

[23] Hamed Jelodar, Yongli Wang, Chi Yuan, Xia Feng, Xiahui Jiang, Yanchao Li, and Liang Zhao. Latent Dirichlet allocation (LDA) and topic modeling: Models, applications, a survey. *Multimedia Tools and Applications*, 78(11): 15169–15211, June 2019. ISSN 1573-7721. doi: 10.1007/s11042-018-6894-4.

[24] Yee Teh, Michael Jordan, Matthew Beal, and David Blei. Sharing Clusters among Related Groups: Hierarchical Dirichlet Processes. In *Advances in Neural Information Processing Systems*, volume 17. MIT Press, 2004.

[25] John Lafferty and David Blei. Correlated Topic Models. In *Advances in Neural Information Processing Systems*, volume 18. MIT Press, 2005.

[26] David M. Blei and John D. Lafferty. Dynamic topic models. In *Proceedings of the 23rd International Conference on Machine Learning*, ICML '06, pages 113–120, New York, NY, USA, June 2006. Association for Computing Machinery. ISBN 978-1-59593-383-6. doi: 10.1145/1143844.1143859.

[27] Chaitanya Chemudugunta, Padhraic Smyth, and Mark Steyvers. Modeling General and Specific Aspects of Documents with a Probabilistic Topic Model. In *Advances in Neural Information Processing Systems*, volume 19. MIT Press, 2006.

[28] Ramesh M. Nallapati, Susan Ditmore, John D. Lafferty, and Kin Ung. Multiscale topic tomography. In *Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '07, pages 520–529, New York, NY, USA, August 2007. Association for Computing Machinery. ISBN 978-1-59593-609-7. doi: 10.1145/1281192.1281249.

[29] Chong Wang, David Blei, and David Heckerman. Continuous Time Dynamic Topic Models. https://arxiv.org/abs/1206.3298v2, June 2012.

[30] Tomoharu Iwata, Shinji Watanabe, Takeshi Yamada, and Naonori Ueda. *Topic Tracking Model for Analyzing Consumer Purchase Behavior*. January 2009.

[31] Arindam Banerjee and Sugato Basu. Topic Models over Text Streams: A Study of Batch and Online Unsupervised Learning. In *Proceedings of the 2007 SIAM International Conference on Data Mining (SDM)*, Proceedings, pages 431–436. Society for Industrial and Applied Mathematics, April 2007. ISBN 978-0-89871-630-6. doi: 10.1137/1.9781611972771.40.

[32] Ian Porteous, David Newman, Alexander Ihler, Arthur Asuncion, Padhraic Smyth, and Max Welling. Fast collapsed gibbs sampling for latent dirichlet allocation. In *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '08, pages 569–577, New York, NY, USA, August 2008. Association for Computing Machinery. ISBN 978-1-60558-193-4. doi: 10.1145/1401890.1401960.

[33] Limin Yao, David Mimno, and Andrew McCallum. Efficient methods for topic model inference on streaming document collections. In *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '09, pages 937–946, New York, NY, USA, June 2009. Association for Computing Machinery. ISBN 978-1-60558-495-9. doi: 10.1145/1557019.1557121.

[34] Matthew Hoffman, Francis Bach, and David Blei. Online Learning for Latent Dirichlet Allocation. In *Advances in Neural Information Processing Systems*, volume 23. Curran Associates, Inc., 2010.

[35] Jon Mcauliffe and David Blei. Supervised Topic Models. In *Advances in Neural Information Processing Systems*, volume 20. Curran Associates, Inc., 2007.

[36] Farial Shahnaz, Michael W. Berry, V. Paul Pauca, and Robert J. Plemmons. Document clustering using nonnegative matrix factorization. *Information Processing & Management*, 42(2):373–386, March 2006. ISSN 0306-4573. doi: 10.1016/j.ipm.2004.11.005.

[37] Shiva Prasad Kasiviswanathan, Prem Melville, Arindam Banerjee, and Vikas Sindhwani. Emerging topic detection using dictionary learning. In *Proceedings of the 20th ACM International Conference on Information and Knowledge Management*, CIKM '11, pages 745–754, New York, NY, USA, October 2011. Association for Computing Machinery. ISBN 978-1-4503-0717-8. doi: 10.1145/2063576.2063686.

[38] Xiaohui Yan, Jiafeng Guo, Shenghua Liu, Xueqi Cheng, and Yanfeng Wang. Learning Topics in Short Texts by Non-negative Matrix Factorization on Term Correlation Matrix. In *Proceedings of the 2013 SIAM International Conference on Data Mining (SDM)*, Proceedings, pages 749–757. Society for Industrial and Applied Mathematics, May 2013. ISBN 978-1-61197-262-7. doi: 10.1137/1.9781611972832.83.

[39] Xiaohui Yan, Jiafeng Guo, Yanyan Lan, and Xueqi Cheng. A biterm topic model for short texts. In *Proceedings of the 22nd International Conference on World Wide Web*, WWW '13, pages 1445–1456, New York, NY, USA, May 2013. Association for Computing Machinery. ISBN 978-1-4503-2035-1. doi: 10.1145/2488388.2488514.

[40] Xiaojun Quan, Chunyu Kit, Yong Ge, and Sinno Jialin Pan. Short and Sparse Text Topic Modeling via Self-Aggregation. In *Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence (IJCAI 2015)*, pages 2270–2276. AAAI Press/International Joint Conferences on Artificial Intelligence, July 2015.

[41] Jianhua Yin and Jianyong Wang. A dirichlet multinomial mixture model-based approach for short text clustering. In *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '14, pages 233–242, New York, NY, USA, August 2014. Association for Computing Machinery. ISBN 978-1-4503-2956-9. doi: 10.1145/2623330.2623715.

[42] Chenliang Li, Haoran Wang, Zhiqian Zhang, Aixin Sun, and Zongyang Ma. Topic Modeling for Short Texts with Auxiliary Word Embeddings. In *Proceedings of the 39th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '16, pages 165–174, New York, NY, USA, July 2016. Association for Computing Machinery. ISBN 978-1-4503-4069-4. doi: 10.1145/2911451.2911499.

[43] Dat Quoc Nguyen, Richard Billingsley, Lan Du, and Mark Johnson. Improving Topic Models with Latent Feature Word Representations. *Transactions of the Association for Computational Linguistics*, 3:299–313, December 2015. ISSN 2307-387X. doi: 10.1162/tacl_a_00140.

[44] Ximing Li, Yue Wang, Ang Zhang, Changchun Li, Jinjin Chi, and Jihong Ouyang. Filtering out the noise in short text topic modeling. *Information Sciences*, 456:83–96, August 2018. ISSN 0020-0255. doi: 10.1016/j.ins.2018.04.071.

[45] Mario Cataldi, Luigi Di Caro, and Claudio Schifanella. Emerging topic detection on Twitter based on temporal and social terms evaluation. In *Proceedings of the Tenth International Workshop on Multimedia Data Mining*, MDMKDD '10, pages 1–10, New York, NY, USA, July 2010. Association for Computing Machinery. ISBN 978-1-4503-0220-3. doi: 10.1145/1814245.1814249.

[46] Henrique F. de Arruda, Luciano da F. Costa, and Diego R. Amancio. Topic segmentation via community detection in complex networks. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, 26(6):063120, June 2016. ISSN 1054-1500, 1089-7682. doi: 10.1063/1.4954215.

[47] Rob Churchill, Lisa Singh, and Christo Kirov. A Temporal Topic Model for Noisy Mediums. In Dinh Phung, Vincent S. Tseng, Geoffrey I. Webb, Bao Ho, Mohadeseh Ganji, and Lida Rashidi, editors, *Advances in Knowledge Discovery and Data Mining*, Lecture Notes in Computer Science, pages 42–53, Cham, 2018. Springer International Publishing. ISBN 978-3-319-93037-4. doi: 10.1007/978-3-319-93037-4_4.

[48] R. Churchill and L. Singh. Percolation-based topic modeling for tweets. *WISDOM 2020 : The 9th KDD Workshop on Issues of Sentiment Discovery and Opinion Mining*, August 2020.

[49] Felipe Almeida and Geraldo Xexéo. Word embeddings: A survey, May 2023.

[50] Yoshua Bengio, Réjean Ducharme, and Pascal Vincent. A Neural Probabilistic Language Model. In *Advances in Neural Information Processing Systems*, volume 13. MIT Press, 2000.

[51] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient Estimation of Word Representations in Vector Space, September 2013.

[52] Jipeng Qiang, Ping Chen, Tong Wang, and Xindong Wu. Topic Modeling over Short Texts by Incorporating Word Embeddings. In Jinho Kim, Kyuseok Shim, Longbing Cao, Jae-Gil Lee, Xuemin Lin, and Yang-Sae Moon, editors, *Advances in Knowledge Discovery and Data Mining*, Lecture Notes in Computer Science, pages 363–374, Cham, 2017. Springer International Publishing. ISBN 978-3-319-57529-2. doi: 10.1007/978-3-319-57529-2_29.

[53] Stefan Bunk and Ralf Krestel. WELDA: Enhancing Topic Models by Incorporating Local Word Context. In *Proceedings of the 18th ACM/IEEE on Joint Conference on Digital Libraries*, JCDL '18, pages 293–302, New York, NY, USA, May 2018. Association for Computing Machinery. ISBN 978-1-4503-5178-2. doi: 10.1145/3197026.3197043.

[54] Ximing Li, Jiaojiao Zhang, and Jihong Ouyang. Dirichlet Multinomial Mixture with Variational Manifold Regularization: Topic Modeling over Short Texts. *Proceedings of the AAAI Conference on Artificial Intelligence*, 33(01):7884–7891, July 2019. ISSN 2374-3468. doi: 10.1609/aaai.v33i01.33017884.

[55] Yishu Miao, Lei Yu, and Phil Blunsom. Neural Variational Inference for Text Processing. In *Proceedings of The 33rd International Conference on Machine Learning*, pages 1727–1736. PMLR, June 2016.

[56] Christopher E. Moody. Mixing Dirichlet Topic Models and Word Embeddings to Make lda2vec, May 2016.

[57] Quoc V. Le and Tomas Mikolov. Distributed Representations of Sentences and Documents, May 2014.

[58] Dimo Angelov. Top2Vec: Distributed representations of topics, August 2020.

[59] Yuan Zuo, Junjie Wu, Hui Zhang, Hao Lin, Fei Wang, Ke Xu, and Hui Xiong. Topic Modeling of Short Texts: A Pseudo-Document View. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '16, pages 2105–2114, New York, NY, USA, August 2016. Association for Computing Machinery. ISBN 978-1-4503-4232-2. doi: 10.1145/2939672.2939880.

[60] Paulo Bicalho, Marcelo Pita, Gabriel Pedrosa, Anisio Lacerda, and Gisele L. Pappa. A general framework to expand short text for topic modeling. *Information Sciences*, 393:66–81, July 2017. ISSN 0020-0255. doi: 10.1016/j.ins.2017.02.007.

[61] Felipe Viegas, Sérgio Canuto, Christian Gomes, Washington Luiz, Thierson Rosa, Sabir Ribas, Leonardo Rocha, and Marcos André Gonçalves. CluWords: Exploiting Semantic Word Clustering Representation for Enhanced Topic Modeling. In *Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining*, WSDM '19, pages 753–761, New York, NY, USA, January 2019. Association for Computing Machinery. ISBN 978-1-4503-5940-5. doi: 10.1145/3289600.3291032.

[62] Felipe Viegas, Washington Cunha, Christian Gomes, Antônio Pereira, Leonardo Rocha, and Marcos Goncalves. CluHTM - Semantic Hierarchical Topic Modeling based on CluWords. In Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel Tetreault, editors, *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8138–8150, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.724.

[63] Adji B. Dieng, Francisco J. R. Ruiz, and David M. Blei. Topic Modeling in Embedding Spaces. *Transactions of the Association for Computational Linguistics*, 8:439–453, July 2020. ISSN 2307-387X. doi: 10.1162/tacl_a_00325.

[64] Adji B. Dieng, Francisco J. R. Ruiz, and David M. Blei. The Dynamic Embedded Topic Model, October 2019.

[65] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is All you Need. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

[66] Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. Improving Language Understanding by Generative Pre-Training. .

[67] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language Models are Unsupervised Multitask Learners. .

[68] Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language Models are Few-Shot Learners, July 2020.

[69] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding, May 2019.

[70] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. RoBERTa: A Robustly Optimized BERT Pretraining Approach, July 2019.

[71] Nils Reimers and Iryna Gurevych. Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks, August 2019.

[72] Laure Thompson and David Mimno. Topic Modeling with Contextualized Word Representation Clusters. https://arxiv.org/abs/2010.12626v1, October 2020.

[73] Jeffrey Pennington, Richard Socher, and Christopher Manning. GloVe: Global Vectors for Word Representation. In Alessandro Moschitti, Bo Pang, and Walter Daelemans, editors, *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, Doha, Qatar, October 2014. Association for Computational Linguistics. doi: 10.3115/v1/D14-1162.

[74] Maarten Grootendorst. BERTopic: Neural topic modeling with a class-based TF-IDF procedure, March 2022.

[75] Xunyu Zhu, Jian Li, Yong Liu, Can Ma, and Weiping Wang. A Survey on Model Compression for Large Language Models, September 2023.

[76] Maarten P. Grootendorst. The Algorithm - BERTopic. https://maartengr.github.io/BERTopic/algorithm/algorithm.html, .

[77] Nils Reimers and Iryna Gurevych. Making Monolingual Sentence Embeddings Multilingual using Knowledge Distillation, October 2020.

[78] Charu C. Aggarwal, Alexander Hinneburg, and Daniel A. Keim. On the Surprising Behavior of Distance Metrics in High Dimensional Space. In Jan Van den Bussche and Victor Vianu, editors, *Database Theory — ICDT 2001*, Lecture Notes in Computer Science, pages 420–434, Berlin, Heidelberg, 2001. Springer. ISBN 978-3-540-44503-6. doi: 10.1007/3-540-44503-X_27.

[79] Kevin Beyer, Jonathan Goldstein, Raghu Ramakrishnan, and Uri Shaft. When Is "Nearest Neighbor" Meaningful? In Catriel Beeri and Peter Buneman, editors, *Database Theory — ICDT'99*, Lecture Notes in Computer Science, pages 217–235, Berlin, Heidelberg, 1999. Springer. ISBN 978-3-540-49257-3. doi: 10.1007/3-540-49257-7_15.

[80] Divya Pandove, Shivan Goel, and Rinkl Rani. Systematic Review of Clustering High-Dimensional and Large Datasets. *ACM Transactions on Knowledge Discovery from Data*, 12(2):16:1–16:68, January 2018. ISSN 1556-4681. doi: 10.1145/3132088.

[81] Michael Steinbach, Levent Ertöz, and Vipin Kumar. The Challenges of Clustering High Dimensional Data. In Luc T. Wille, editor, *New Directions in Statistical Physics: Econophysics, Bioinformatics, and Pattern Recognition*, pages 273–309. Springer, Berlin, Heidelberg, 2004. ISBN 978-3-662-08968-2. doi: 10.1007/978-3-662-08968-2_16.

[82] Leland McInnes, John Healy, and James Melville. UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction, September 2020.

[83] Leland McInnes, John Healy, and Steve Astels. Hdbscan: Hierarchical density based clustering. *The Journal of Open Source Software*, 2(11):205, March 2017. ISSN 2475-9066. doi: 10.21105/joss.00205.

[84] Mebarka Allaoui, Mohammed Lamine Kherfi, and Abdelhakim Cheriet. Considerably Improving Clustering Algorithms Using UMAP Dimensionality Reduction Technique: A Comparative Study. In Abderrahim El Moataz, Driss Mammass, Alamin Mansouri, and Fathallah Nouboud, editors, *Image and Signal Processing*, Lecture Notes in Computer Science, pages 317–325, Cham, 2020. Springer International Publishing. ISBN 978-3-030-51935-3. doi: 10.1007/978-3-030-51935-3_34.

[85] Thorsten Joachims. A Probabilistic Analysis of the Rocchio Algorithm with TFIDF for Text Categorization. In *Proceedings of the Fourteenth International Conference on Machine Learning*, ICML '97, pages 143–151, San Francisco, CA, USA, July 1997. Morgan Kaufmann Publishers Inc. ISBN 978-1-55860-486-5.

[86] Maarten Grootendorst. MaartenGr/KeyBERT, February 2024.

[87] Explosion/spaCy: Industrial-strength Natural Language Processing (NLP) in Python. https://github.com/explosion/spaCy/tree/master, .

[88] Jonathan Chang, Sean Gerrish, Chong Wang, Jordan Boyd-graber, and David Blei. Reading Tea Leaves: How Humans Interpret Topic Models. In *Advances in Neural Information Processing Systems*, volume 22. Curran Associates, Inc., 2009.

[89] David Newman, Jey Han Lau, Karl Grieser, and Timothy Baldwin. Automatic Evaluation of Topic Coherence. In Ron Kaplan, Jill Burstein, Mary Harper, and Gerald Penn, editors, *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 100–108, Los Angeles, California, June 2010. Association for Computational Linguistics.

[90] Silvia Terragni, Elisabetta Fersini, Bruno Giovanni Galuzzi, Pietro Tropeano, and Antonio Candelieri. OCTIS: Comparing and Optimizing Topic models is Simple! In Dimitra Gkatzia and Djamé Seddah, editors, *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: System Demonstrations*, pages 263–270, Online, April 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.eacl-demos.31.

[91] Francesco Archetti and Antonio Candelieri. *Bayesian Optimization and Data Science*. SpringerBriefs in Optimization. Springer International Publishing, Cham, 2019. ISBN 978-3-030-24493-4 978-3-030-24494-1. doi: 10.1007/978-3-030-24494-1.

[92] B. G. Galuzzi, I. Giordani, A. Candelieri, R. Perego, and F. Archetti. Hyperparameter optimization for recommender systems through Bayesian optimization. *Computational Management Science*, 17(4):495–515, December 2020. ISSN 1619-6988. doi: 10.1007/s10287-020-00376-3.

[93] Jasper Snoek, Hugo Larochelle, and Ryan P Adams. Practical Bayesian Optimization of Machine Learning Algorithms. In *Advances in Neural Information Processing Systems*, volume 25. Curran Associates, Inc., 2012.

[94] Sentence-transformers/all-mpnet-base-v2 · Hugging Face. https://huggingface.co/sentence-transformers/all-mpnet-base-v2, January 2024.

[95] Microsoft/mpnet-base · Hugging Face. https://huggingface.co/microsoft/mpnet-base, .

[96] Maarten P. Grootendorst. 1. Embeddings - BERTopic. https://maartengr.github.io/BERTopic/getting_started/embeddings/embeddings.
.

[97] Hugging Face – The AI community building the future. https://huggingface.co/, January 2024.

[98] GitHub: Let's build from here. https://github.com/, .

A Appendix

You may provide any type of material as appendix to your project proposal. Typical appendices include additional details about the methodology, further pilot studies for illustration and demonstration of feasibility, images and results that were created, pointers to code (or pseudo-code itself), pointers to data, etc. There is no limit in the length of the appendices. For example, this appendix contain Table 3, which has information that are not relevant but shows how to use a table here.

Table 3: Some results of something. It is recommend not to try to understand it.

				Median		Maximum	
Problem	Description	Max Value	Nodes	Memory	Time(s)	Memory	Time(s)
MINAP	naive Bayes w/ random params	10^6	50	59	0.06	84	0.08
			100	125	0.198	200	0.285
			200	396	1.328	1238	1.893
			300	1103	2.793	20863	9.893
MAP	naive Bayes w/ random params	10^6	50	5	0.01	7	0.015
			100	5	0.017	6	0.023
			200	5	0.04	7	0.047
			300	5	0.043	7	0.049
MINAP	partition problem	10^4	10	512	0.034	512	0.039
			20	91857	11.553	100842	17.42
			30	236979	77.09	264638	82.81
		10^5	10	512	0.036	512	0.045
			20	347065	27.599	372670	31.10
			30	2046264	532.318	2237859	586.4
		10^6	10	512	0.035	512	0.038
			20	501347	34.672	510413	38.13
			30	$>$ 10Mln	$>$ 600	$>$ 10Mln	$>$ 600
MINAP	random struct. and parameters	10^6	50	57	0.046	101	0.073
			100	143	0.21	197	0.326
			200	417	1.288	713	1.761
			300	1129	2.509	10403	14.53
MAP	random struct. and parameters	10^6	50	5	0.009	7	0.014
			100	5	0.018	6	0.023
			200	5	0.042	7	0.047
			300	6	0.049	7	0.061

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

Nulla malesuada porttitor diam. Donec felis erat, congue non, volutpat at, tincidunt tristique, libero. Vivamus viverra fermentum felis. Donec nonummy pellentesque ante. Phasellus adipiscing semper elit. Proin fermentum massa ac

quam. Sed diam turpis, molestie vitae, placerat a, molestie nec, leo. Maecenas lacinia. Nam ipsum ligula, eleifend at, accumsan nec, suscipit a, ipsum. Morbi blandit ligula feugiat magna. Nunc eleifend consequat lorem. Sed lacinia nulla vitae enim. Pellentesque tincidunt purus vel magna. Integer non enim. Praesent euismod nunc eu purus. Donec bibendum quam in tellus. Nullam cursus pulvinar lectus. Donec et mi. Nam vulputate metus eu enim. Vestibulum pellentesque felis eu massa.

Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula.