A KNOWLEDGE-GRAPH BASED GRAPH NEURAL NETWORK MODEL TO IDENTIFY TOPICS IN SHORT TEXTS

by

Abdullah Atakan Güney

B.S., Computer Engineering, Boğaziçi University, 2018

Submitted to the Institute for Graduate Studies in
Science and Engineering in partial fulfillment of
the requirements for the degree of
Master of Science

Graduate Program in Computer Engineering

Boğaziçi University

2022

ACKNOWLEDGEMENTS

Special thanks to my supervisor, Assist. Prof. Suzan Üsküdarlı for her assistance, patience, and support.

In addition, I would like to thank you my parents and wife for their support and understanding during our work.

This work is supported by the Turkish Directorate of Strategy and Budget under the TAM Project number 2007K12-873.

ABSTRACT

A KNOWLEDGE-GRAPH BASED GRAPH NEURAL NETWORK MODEL TO IDENTIFY TOPICS IN SHORT TEXTS

Topic models are probabilistic generative models used to analyze a collection of documents. People have leveraged topic models for many years to extract hidden structures from documents. However, classical topic models such as Latent Dirichlet Allocation (LDA) have issues with short texts typical in user-generated social media content. Due to the limited context of short texts, they fail to learn interpretable topics from extensive vocabularies with the bag of word representations that do not represent them well. This thesis proposes a topic model based on Graph Neural Networks (GNN) where documents are represented as graphs with entity-specific relations using Wikidata as a knowledge graph. A graph attention network learns the embeddings of these documents whose outputs are passed to the probabilistic generative topic model Entity Embedded Topic Modeling (EETM) as probability distribution parameters to yield the topics. We evaluate our model with various short text collections fetched from Twitter related to politics, sports, pandemics, and trending news events. We provide a detailed discussion regarding our observations related to the learned embeddings and qualities of topics resulting from our model.

ÖZET

KISA YAZILAR İÇERİSİNDEKİ KONULARI TANIMLAMAK İÇİN BİLGİ GRAFİĞİ TABANLI ÇİZGE SİNİR AĞI MODELİ

Konu modelleri, bir belge koleksiyonunu analiz etmek için kullanılan olasılıksal üretken modellerdir. İnsanlar, belgelerden gizli yapıları çıkarmak için uzun yıllardır konu modellerinden yararlandı. Ancak, Gizli Dirichlet Tahsisi (LDA) gibi klasik konu modelleri, kullanıcı tarafından oluşturulan sosyal medya içeriğinde tipik olan kısa metinlerle ilgili sorunlar yaşar. Kısa metinlerin sınırlı bağlamı nedeniyle, onları iyi temsil etmeyen kelime temsilleri çantasıyla geniş kelime dağarcığından yorumlanabilir konular öğrenemezler. Bu tez, belgelerin, bilgi grafiği olarak Wikidata kullanılarak, varlığa özel ilişkilere sahip çizgeler olarak temsil edildiği, Çizge Sinir Ağlarına (GNN) dayalı bir konu modeli önermektedir. Bir çizge dikkat ağı, bu belgelerin yerleştirmelerini ve kendi çıktılarını, olasılıksal üretken konu modeli olan Varlık Gömülü Konu Modellemesi'ne (EETM), konuları elde edebilmesi için, olasılıksal dağılım parametreleri şeklinde geçirerek öğrenir. Modelimizi Twitter'dan siyaset, spor, pandemi ve trend olan haberlerle ilgili çeşitli kısa metin koleksiyonları ile değerlendirdik. Modelimizden sağladığı konuların öğrenilen yerleştirmeleri ve nitelikleri ile ilgili gözlemlerimizle ilgili ayrıntılı bir tartışma sunduk.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

1. INTRODUCTION

With the growth of social media platforms, it became more valuable to understand hidden structures of user-generated data of short texts such as comments, reviews, and tweets. For years, topic models have been leveraged to extract hidden structures from documents. The most common approach in topic modeling is Latent Dirichlet Allocation (LDA) [1], a probabilistic generative model. The LDA and its extensions express topics as a mixture of words and documents as a mixture of topics. Since they emerged, the LDA-based models successfully analyzed text data [2], [3], [4].

Nevertheless, they have shortcomings when applied to short texts such as social media content. In particular, they represent documents with well-known Bag of Words (BoW) representation which is not expressive in short texts. Moreover, BoW representation has the main drawback. It suffers from conditionally independence assumption among words. Since it assumes all the words of a document are conditionally independent, it ignores semantic relationships between words of a document. Also, they prune the large vocabularies to fit their models, which causes losing information from the data; that is another issue on social media texts because a tremendous number of users yields an extensive vocabulary.

In general, topic modeling is a Natural Language Processing task that identifies the semantic concepts from the observed documents in terms of their pieces. Moreover, each topic or semantic concept consists of words, entities, or other types of pieces that form the documents. For instance, let us have a set of two tweets considered documents.

(i) Lakers, LeBron James focusing on Chris Paul pass, ready to present up three gamers
(ii) Is Lebron James still the best player in the world?

If we represent each tweet in words contained in them, an example of the two topics extracted from this set of documents will be:

(i) Lakers, Lebron, James, best
(ii) Chris Paul, pass, Lebron

Each word contributes to each topic to some extent, while a topic is a distribution over words. Furthermore, each topic contributes to each document because a document is distribution over topics. Note that a semantic concept corresponds to a set of words in this example.

In this work, we address two problems. First, we aim to solve the problem of having limited context for a short document. Secondly, we use an embedding vector based solution to overcome the difficulty of handling giant vocabularies. We introduce a knowledge graph based solution to enrich the context of a document and adapt an embedding vector based learning of topics.

In [5], documents are represented as graphs to overcome shortcomings of the BoW representation. However, they create a document graph based on the distance between words. Although they preserve the order information of words, they also suffer from representing documents with short texts because the document graph will be very sparse. In [6], a topic model based on word embedding vectors proposed, and they overcome the issue of an extensive vocabulary, but they suffer from BoW representation of documents.

This thesis proposes a topic modeling technique based on distributed representations of entities of a knowledge graph to overcome both issues. As the knowledge graph, we have leveraged Wikidata [7], a collaboratively edited knowledge graph hosted by Wikimedia Foundation [7]. We aimed to discover external knowledge in the topic modeling of short texts and leveraged Graph Attention Network (GAT) [8] architecture for embedding external information.

First, we extracted entities from documents and linked them to the Wikidata. Afterward, we extracted the paths on Wikidata between those entities. Then, inspired by [9], we form a joint graph called a working graph for each document that contains document entity nodes and the context node along with the relevance score for each node in the graph. In our working graph, the nodes represent the entities and context-node. Besides, the relationships represent the Wikidata relationships between entities and the co-occurrence relationship we added between each entity pair of the set entities included in the original document. We fed the constructed graphs into the GAT architecture implemented, same with [9]. It fulfills our needs of representing both node and edge types in the attention mechanism.

We integrated the GAT architecture mentioned above with our topic modeling structure. Although it is similar to LDA, the model does not assume that the words come from a categorical distribution; instead, the model samples words from a distribution calculated by a function of topic and entity embedding vectors and estimates the marginal likelihood of entities similar to [6]. The difference is that instead of Multi Layer Perceptron (MLP), we leverage our GAT architecture in likelihood calculation. In addition, instead of using words as a source of documents like the Embedded Topic Model [6], we used entities. Therefore, we named our model as Entity Embedded Topic Model.

For evaluation, we collected data sets from the most famous microblogging service, Twitter. In particular, we selected our data sets of tweets from various subjects, namely Black Lives Matter (BLM), U.S. Capitol Attacks on January 6 (January 6), Covid 19 pandemic (Covid 19), three populist leaders around the world (Populist Leaders), and National Basketball Association (NBA).

As the literature indicates, evaluating topic models is a difficult task, requiring own research. However, we evaluated our model in quantitative and qualitative manners. We measured topic coherence and topic diversity metrics for quantitative analysis. In addition, we have examined our extracted entities for each topic to see if

they are interpretable. Moreover, to understand the harmoniousness between learned embedding vectors for entities and topics, we have mapped them into two-dimensional space and visualized how they are close. We also examined the expressiveness of the learned entity embedding vectors by utilizing them in representing the tweets and using the representations in a clustering task.

Overall, we introduced a new document representation technique for short texts that enriches the context of a short text beyond its words. Moreover, we preserved the relationships between the pieces of a document. We represented each document as a graph. We overcome the sparsity challenge of these small graphs by enriching them with new nodes introduced from an external source, a knowledge graph. Another challenge is to extract information from the knowledge graph. We proposed two different approaches to solve this issue. First, we created a subgraph of the knowledge graph, and secondly, we used the query service of the external knowledge graph. Whereas the former approach is applicable for all knowledge graphs, the latter is only available if such a service exists. Moreover, to the best of our knowledge, our work is the first one that models topics as a distribution of knowledge graph entities.

The following chapters of this thesis are; Chapter 2 explains the related work in both topic modeling and graph representation areas, Chapter 3 shows the necessary background to understand the thesis. Chapter 4 defines our approach in detail. Chapter 5 clarifies the implementation details. Chapter 6 presents our experiments and evaluations, and Chapter 7 and Chapter 8 describe findings, possible future work, and conclusion remarks.

2. RELATED WORK

We review the literature in various aspects. First, we investigated works that leverage knowledge graphs in various NLP tasks. Second, we analyzed knowledge graph embedding methods. Third, we captured previous works which leverage many GNN architectures in various NLP tasks, including topic modeling. Finally, we revisited topic modeling works, explicitly using embedding vectors.

2.1. Knowledge Graphs Used in NLP

In [10], language models are analyzed if they represent relational knowledge or not. They prepare a data set by using various knowledge graphs.

In [11], a knowledge graph is utilized as an external source to enrich their question-answer corpus. After enriching the question, they use a pre-fine-tuned model to predict the answer. Although they use a knowledge graph, they do not reflect the relationships.

In [12], a pre-training approach that exploits a knowledge graph is proposed. They extend their training examples by using the knowledge graph. Similar to previous work, they do not represent relationships between entities.

In [13], it is aimed to solve knowledge base completion task. So, they leverage a knowledge graph with a language model to enrich knowledge bases.

2.2. Knowledge Graph Embedding Methods

Before embedding vector approaches that leverage pre-trained language models and complex deep architectures, more traditional Knowledge Graph Representation Learning methods that leverage techniques like matrix factorizations are used in the literature. Complex Embeddings [14], TransE [15] learn knowledge graph embedding

vectors by matrix factorization techniques. Instead of leveraging the pre-trained language model, these models directly learn the entities' representation from the knowledge graph.

In [16], a Graph Attention Mechanism is proposed to model a knowledge graph in a recommendation system. It propagates information through Graph Attention Layers similar to TransE [15].

In [9] and [17], a method that leverages an external knowledge graph is proposed to enrich the representation of the question. They both represent questions as graphs. In [9], a relevant scoring mechanism is added which measures how the external knowledge is related to the existing context. In our work, we used the same technique to represent our documents as [9] did.

2.3. GNN Architecture in NLP

In [18], Neural Topic Modeling, a Graph Convolutional Network based topic modeling, is introduced. The graph structure is constructed by the documents' shared words in that model. Hence the constructed graph reflects the cross relationships among documents. A GCN-based model learns the representation of the constructed graph. This approach reflects cross relationships among documents very well. However, it also suffers from not connecting words within the document. Also, the short context problem remains as well.

Furthermore, in [5], a topic model in which a graph of words represents documents is learned. Then, with the relationships of the words, topics' distributions over words are learned by GCN architecture. The relationships represent how two words are close within a document in this work. Although this work represents relationships within a document, it only reflects the closeness of two words. Hence, the other semantic relationships are not reflected via this approach. Moreover, it does not also provide a solution to the short context.

2.4. Embeddings in Topic Modeling

For years, the most common approach used and adapted to solve topic modeling is Latent Dirichlet Allocation [1]. The successors of this approach did not show significant performance improvements. Still, our topic model adapted the idea from Latent Dirichlet Allocation [1]. Based on LDA, in [6], a topic modeling technique that leverages the distributional representation of words, the word embedding vectors, and the distributional representation of the topics in the same vector space is proposed. Moreover, they merged the LDA and embedding vector approach inspired by Continues Bag of Words (CBoW) architecture of [19].

In [20], also the topics are modeled as a distribution over entities. However, instead of leveraging embedding vectors, that approach implements the LDA [1] with entities.

In this work, we combine two ideas from two independent works, namely topic modeling with embedding vectors [6] and leveraging external knowledge graph entities to analyze natural texts [9].

As far as we know, no previous work leverages Graph Attention Networks [8] for topic modeling. Nevertheless, related works use a similar architecture for Graph Representation Learning.

3. BACKGROUND

3.1. Graph Neural Networks

Recent advances in deep learning techniques have had highly favorable impacts on various NLP tasks such as *question answering* and *neural machine translation*. Mainly, the data in these tasks are represented in the Euclidean space. However, various kinds of data sets are naturally in non-Euclidean spaces, and *graphs* naturally can represent these data sets and their complex inner relationships. For instance, *drug-drug* and *drug-protein* relationships used in bioinformatics domain.

Many studies address this problem by extending current deep learning techniques into graphs. In [21], *Graph Neural Network* architecture is introduced which is an extension of a *Recurrent Neural Network* architecture for graph structured data sets. These GNNs constantly exchange node information until an equilibrium called the message passing mechanism. Furthermore, this work affected its successors.

Afterward, *Convolutional Neural Networks* are adapted by GNNs. There are two types of Convolutional GNNs. First one is *spectral* ones. These network architectures have a solid mathematical foundation. They operate graph convolution on graph Laplacian [22] defined as $\mathcal{L} = \mathbb{I} - \mathbb{D}^{\frac{-1}{2}}\mathbb{A}\mathbb{D}^{\frac{-1}{2}}$ where $\mathbb{D}$ is the diagonal matrix of node degrees and $\mathbb{A}$ is the adjacency matrix. A *spectral* GCN is proposed in [23] for node classification problem. However, since it uses the graph Laplacian, this approach has a generalization problem. The model learned on a particular structure, i.e., the graph Laplacian, cannot be directly applied to other structures.

Second type of Convolutional GNNs are *spatial* Convolutional GNNs. Similar to *Recurrent Neural Network* based GNNs, the graph convolution function computes the convolution on the localized neighborhood of nodes. In other words, the model calculates a node's representation by convolving its neighbors' representations as showed

in the Figure 3.1. Spatial Convolutional GNNs could represent typical image convolution [24] operation as a particular case of it. In this case, pixels of an image correspond to nodes, and each pixel is connected to its nearby neighbor pixels. Figure 3.2 shows an example graph representation of an image.

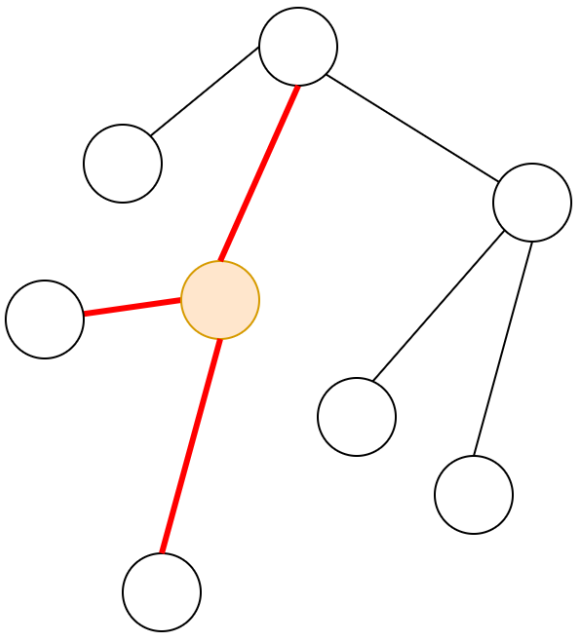

Figure 3.1. Graph convolution representation on an arbitrary graph.

In regular convolutional GNNs, the model aggregates the features of a node's neighbors by taking the weighted average of the values with calculated weights such as node degrees. In [8], a new mechanism is introduced for GNNs to aggregate the node features. Instead of taking a mean or weighted average, the proposed network architecture learns the importance of neighbors by *attention* mechanism.

As the attention mechanism for a sequence-based task is introduced in [25], it became a de facto approach in many sequence-based studies. Recurrent Neural Networks or Convolutional Neural Networks could benefit from the attention mechanism in many tasks such as learning sentence representation [26]. Moreover, in [27], it is showed that not just using the attention mechanism along with another architecture could improve it, but the attention mechanism could be a model architecture solely.

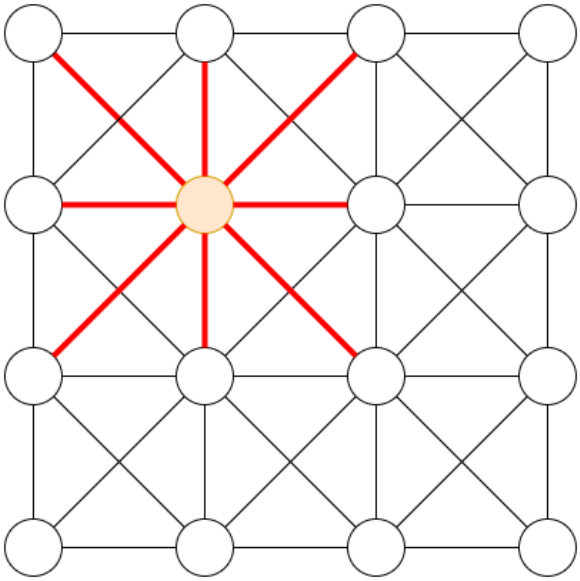

Figure 3.2. This figure shows graph representation of a 2D image and corresponding convolution operation for the highlighted node.

This approach influenced by [8] and Figure 3.3 shows how the information propogated by the graph attention mechanism.

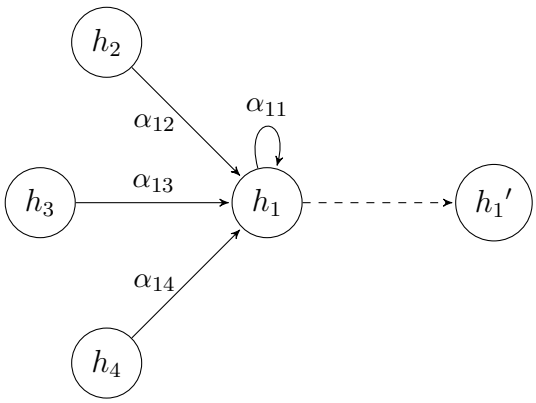

Figure 3.3. Information propagation from the neighborhood of a node in graph attention mechanism.

3.1.1. Message Passing

In GAT architecture, the critical part is the message-passing mechanism responsible for propagating the information from a node's local neighbors to itself. For a given graph $G = (V, E)$, in a GAT layer, a node's representation vector at iteration $t + 1$, $\vec{h}_n^{t+1}$, is updated by

$$\vec{h}_n^{t+1} = f(\sum_{i \in \mathcal{N}(n)} \alpha_i \cdot \vec{m}_i^t) + \vec{h}_n^t. \tag{3.1}$$

Here, $\vec{m}_i^t$ is message or information from local neighborhood,$\mathcal{N}$ of node n. Local neighborhood, $\mathcal{N}$ defined as

$$\mathcal{N}(n) = \{i | \exists (i, n) \in E \vee i = n\}. \tag{3.2}$$

Following a similar idea, in [9] a new message passing mechanism that considers node types and relation types of graphs is proposed. They achieved this by integrating node and relation types as one-hot vectors into the attention score, α calculation.

3.2. Twitter API v2 and Tweets

In our work, we leverage the set of *tweets* as our model's input data set. A tweet is a short text containing 280 characters and shared in public or among friends through the social media application TWITTER. TWITTER provides an API to developers and researchers from which consumers can fetch the tweets shared on the application. A tweet has various fields provided by Twitter API. In this work we fetch tweets using the Twitter API v2 [28]. A set of tweets is collected by providing queries and indicating the desired fields, namely:

- id: the unique identifier of this Tweet, type is *string*
- created_at: creation time of the Tweet, type is a date (ISO 8601)
- author_id: the unique identifier of this user, type is a string

- text: the content of the Tweet, type is a string
- entities: contains details about text that has special meaning in a Tweet. It might contain *annotations*, *hashtags*, *urls*, *mentions*, *cashtags*, type is object
- context_annotations: contains context annotations for the Tweet, type is array

In this work, we leveraged the Twitter API v2 [28] as our source of data. Twitter has released the Twitter API for developers and researchers to retrieve and analyze the set of tweets. It has various certain features to fetch desired set of data. The API serves both historical and recent sets of tweets.

For academic usage, Twitter provides Academic License to use some premium features. We leveraged this academic license to fetch our data sets. Recently, they released the API v2. They fixed bugs from API v1 and added new features and structures to existing ones.

3.2.0.1. Context Annotations. For our work and research, the most important new feature is *context annotations*. The context annotations consist of the entities related to the Tweet, but it does not require to be within the tweet. For instance, the tweet's author might not be in the tweet's text, but it is in the context annotations. Another example would be the mentioned people. Twitter developed a structural analysis tool for annotations in a tweet. This analysis feature puts the annotations of the named entities in specific categories defined by the company. According to their documentation, they have 50+ number of categories for the context annotations, but the exact number is unknown.

3.2.0.2. Entity Annotations Sourced by Twitter API. In addition to context annotations, they also provide legacy entity annotations. The main difference between context annotations and entity annotations is that letters should appear in the text, but the former should not. For instance, the tweet's author might not appear in the text but in the set of context annotations.

```
[{"start": 59,
  "end": 61,
  "probability": 0.8278,
  "type": "Organization",
  "normalized_text": "NBA"}]
```

Figure 3.4. Example entity annotation from Twitter API.

On the other hand, if the author is not in the text, entity annotations do not include it. An entity annotation item consists of *confidence score*, and *start and end* fields. Entity annotations provided by Twitter API are annotations contained within the Tweet text and they have following types.

- Person
- Place
- Product
- Organization
- Other

As a concrete example, consider that we have the following tweet:

#NBA Legend #PatrickEwing thinks that big guys in this NBA don't play the game how is supposed to

#nbayoungboy #nbabasketball #nbanews #nbahighlights #nbavote #nba #basketball #nbabasketball https://t.co/xjgb9R8g1c

The example of entity annotations is showed in the Figure 3.4, and an example of the context annotations depicted in the Figure 3.5.

```
[{
  "domain": {
    "id": "3",
    "name": "TV Shows",
    "description": "Television shows from around the
       world"
  },
  "entity": {
    "id": "10000607734",
    "name": "NBA Basketball",
    "description": "All the latest basketball action
       from the NBA."
  }
}]
```

Figure 3.5. Example context annotations from Twitter API.

3.3. Knowledge Graphs

A *graph*, $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, comprise a set of nodes, $\mathcal{V}$ and a set of edges, $\mathcal{E}$, where an edge denotes the relationship going from $u \in \mathcal{V}$ to $v \in \mathcal{V}$, that is represented by $(u, v) \in \mathcal{E}$. If the relationship is *symmetric*, i.e. $(u, v) \in \mathcal{E} \rightarrow (v, u) \in \mathcal{E}$, the graph is called undirected. In our work, we will mainly be dealing with *directed* graphs.

If a graph contains different relationships, i.e., they have various types, the graph is called multi-relational. Formally, a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ has a set of edges, $\mathcal{E} = \{(u, \tau, v) | u, v \in \mathcal{V} \wedge \tau \in \mathcal{R})\}$, $\mathcal{R}$ is the set of relationships. For instance, in a graph that depicts *drug* - *drug* relationships, we might want to differ the relations which represent side effects if two related drugs are taken together [29].

In the literature, there is not a particular definition of *what is a knowledge graph.* Most of the works in this area have defined their knowledge graph structure. There is also independent research that has been done only on the definition of a knowledge graph [30]. This work defines a knowledge graph as follows:

Definition 3.3.1. A knowledge graph, $\mathcal{KG}$ is a multi relational directed graph which means, it consists of $(\mathcal{V}, \mathcal{E})$ where $\mathcal{V}$ is the set of nodes and $\mathcal{E}$ are the relationships among them. $\mathcal{E} = \{(u, \tau, v) | u, v \in \mathcal{V} \wedge \tau \in \mathcal{R}\}$ where $\mathcal{R}$ is the set of types of edges.

An entity is a thing with distinct and independent existence in the real world. An entity in a *knowledge graph* is a thing that describes a real-world entity. Thus, the nodes of a *knowledge graph* represent entities from the real world.

Object relationships or object properties are relationships between two entities within the $\mathcal{KG}$. Another concept that is worth mentioning regarding knowledge graphs is property paths. A property path is a simple path between two nodes of the graph. As a concrete example, if a knowledge graph contains the following triples,

(i) $s_1 \xrightarrow{p_1} s_2$
(ii) $s_2 \xrightarrow{p_2} s_3$

a property path between s_1 and s_3 is $s_1 \xrightarrow{p_1} s_2 \xrightarrow{p_2} s_3$.

3.4. TagMe

TAGME [31] is a tool for entity linking. Entity linking is extracting and mapping entities with external knowledge within a text. TAGME links entities onto Wikipedia articles.

A TAGME [32] annotation object has the following structure:

- spot: the text piece of annotation, type is a string
- start: the position where the spot starts, type is an integer
- link_probability: the probability of the annotation, type is float
- rho: rho value of the annotation, type is float
- end: the position where the spot ends, type is an integer

TAGME returns a value associated with each annotation called ρ, which predicts the goodness of the annotation concerning other entities within the text [31]. Additionally, it also returns *link probability* which measures the reliability of the corresponding substring as a significant mention. We have used both values in post-processing the returned annotations to filter our set of annotations. As a concrete example, if we assume to have the following tweet as input:

not sure if much more needs to be added here, maybe just that the 90's were wild #nbayoungboy #nbabasketball #nbanews #nbahighlights #nbavote #nba #basketball #nbabasketball #pippen #madonna

```
POST "https://tagme.d4science.org/tagme/tag"
```

Figure 3.6. TAGME API query.

We make the request like in the Figure 3.6 for each tweet, and the json data of the request is depicted in the Figure 3.7. As the response, the API returns annotations. The Figure 3.8 shows an example TAGME annotation.

```
{
  "text": "not sure if much more needs to be added
    here, maybe just that the 90's were wild \#
    nbayoungboy \#nbabasketball \#nbanews \#
    nbahighlights \#nbavote \#nba \#basketball \#
    nbabasketball \#pippen \#madonna",
  "lang": "en",
  "include_all": "true",
  "tweet": "true",
  "include_categories": "true",
  "gcube-token": "GCUBE-TOKEN"
}
```

Figure 3.7. Sample post data for TAGME API.

```
{"spot": "sure",
  "start": 6,
  "link_probability": 0.0038795273285359144,
  "rho": 0.07165562361478806,
  "dbpedia_categories": ["1994 singles", "Take That
    songs", "Songs written by Gary Barlow", "Songs
    written by Robbie Williams", "Songs written by
    Mark Owen"],
  "end": 10,
  "id": 9892133,
  "title": "Sure (Take That song)"}
```

Figure 3.8. Example entity annotation from TAGME API.

3.5. SparQL

For years, the language of querying a database has been *Structured Query Language* (SQL). This language is *de facto* standard for the industry and the academic world. It has various derivations in different fields. Although it is for relational databases, most non-relational databases also adapt similar query languages for their databases.

SPARQL [33] is the recommendation of W3C foundation for querying language for Resource Description Framework(RDF). RDF is a standard introduced by W3C for representing information on the Web. The information contained on the Web is complex and hard to maintain. Hence, the SPARQL [33] is to be able to run in a distributed manner, and it is robust to change of schema of the information.

In this work, we use Wikidata's SPARQL [33] Query Engine to fetch desired information from Wikidata. Using this engine helped us fetch and store data from whole Wikidata.

We implemented our property path extraction logic between two entities of a given tweet based on SPARQL. Moreover, we compared these results with the approach based on property path extraction with a sub-knowledge graph of Wikidata.

3.6. PyTorch

Recent advances in deep learning increased the attention of new libraries to ease the development process of the models and increase their performance. Besides, Python has become the standard programming language for machine learning and deep learning. Among various deep learning libraries, PyTorch [34] is the most pythonic deep learning library. The library's running principles match with the programming language. Moreover, it does not use another language as its backbone.

The main feature of deep learning libraries is that they help compute automatic differentiation. We have selected PyTorch as our deep learning library for its popularity and performance. The library's popularity contributed to it because it has a vast amount of support from its community, and the community implemented various extra ready-to-use libraries on top of Python. For our purposes, we leverage the torch-geometric [35] library developed on top of PyTorch to write Graph Neural Networks and train them. It reduced considerable overhead from our implementation. The library provides information propagation, i.e., message-passing methods needed to implement Graph Attention Network architecture.

3.7. Evaluation of Topics

Evaluation of extracted topics requires its own research. In the literature, we have observed different works which address this issue, e.g., [36], and [37]. *Topic coherence* and *topic diversity* are two metrics to measure the quality of an extracted topic. In this work, we also leveraged these two metrics.

3.7.1. Topic Coherence

Topic coherence is a metric that scores a single topic. Various works use this metric to measure how interpretable a topic is, e.g., [6]. There are various kinds of *topic coherence* calculations used in previous works. The featured ones are followings:

- C_V **Coherence Score**; One of the most popular topic coherence metrics, based on created content vectors of words using their co-occurrences.
- C_{UMass} ; it estimates the probability of two words appearing together in the corpus by their frequencies with the formula

$$C_{UMass} = \frac{2}{N \cdot (N-1)} \sum_{i=2}^{N} \sum_{j=1}^{i-1} \log \frac{P(w_i, w_j) + \epsilon}{P(w_j)}. \tag{3.3}$$

- C_{UCI}; it is similar to C_{UMass} but instead of searching for two words appearing together, it estimates the join proability of two words using sliding windows.
- $C_{word2vec}$; this metric calculates the *intra* and *inter* topic similarities, and coherence score based on them.

3.7.2. Topic Diversity

Topic Diversity is another metric for measuring the quality of a given topic. This metric measures how much a topic includes different words. In general, taking the top N words from the topic calculates the unique words' ratio.

In our work, we have used C_{UMass} as our *topic coherence* metric and *topic diversity*.

3.8. Topic Modeling

Topic Modeling is a statistical approach to express a collection of documents in terms of themes contained in it. The topic modeling process is all about extracting meaningful sets of words that identify the collection. It is, moreover, clustering the documents within the collection under those sets of words. In general, the sets mentioned above of words are called topics.

3.8.1. Latent Dirichlet Allocation

The most famous proposal in the topic modeling field is *Latent Dirichlet Allocation* which is proposed in [1]. Informally, the idea is to model the document generation process by leveraging latent variables. These latent variables are learned as topics. So, each document is assumed to be a distribution over topics, and each topic is assumed to be a distribution over vocabulary words. This generative process for each document, d in the corpus is as follows:

(i) Choose a topic proportion $\theta \sim Dirichlet(\alpha_\theta)$

(ii) For each word w_i in a document d

  (a) Choose a topic $z_i \sim Cat(\theta)$

  (b) Choose the word $w_i \sim Cat(\beta_{z_i})$ where $\beta_{z_i} \sim Dirichlet(\alpha_\beta)$.

Here, α_θ and α_β are hyperparameters tuned during training. Those parameters are passed to the Dirichlet probability distribution function to sample θ and β_{z_i}. θ, and β_{z_i} are parameters for Categorical probability distribution functions, abbreviated as Cat. Moreover, θ is used to sample a topic for each word, whereas β_{z_i} is used to sample the word for a given topic z_i.

In training, the inferential problem that it needs to solve is that of computing the posterior distribution of the hidden variables given a document;

$$p(\theta, \mathbf{z}|\mathbf{w}, \alpha_\theta, \alpha_\beta) = \frac{p(\theta, \mathbf{z}, \mathbf{w}|\alpha_\theta, \alpha_\beta)}{p(\mathbf{w}|\alpha_\theta, \alpha_\beta)}. \tag{3.4}$$

LDA is highly elaborated in the literature for different subjects. The idea is applied in various fields such as bio-informatics, sequence analysis, and natural language processing.

3.8.2. Embedded Topic Modeling

As an extension to the famous LDA approach, in [6], a pretty similar model with a big difference is proposed. Since the foundation of the word embedding vectors by [19], the idea has been highly adopted in many research fields. In [6], the embedding vectors are learned during a generative process similar to LDA. Moreover, the proposed *embedded topic modeling* is also learning the embedding vectors for topics.

The generative process for each document d in a given corpus proposed by ETM as follows:

(i) Choose a topic proportion $\theta \sim LogisticNormal(0, \mathbf{I})$
(ii) For each word w_i in a document d
  (a) Choose topic assignment $z_i \sim Cat(\theta)$
  (b) Choose the word $w_i \sim softmax(\rho^\top \alpha_{z_i})$.

Note that θ is drawn as follows:

(i) Choose $\delta \sim \mathcal{N}(0, \mathbf{I})$, where $\mathcal{N}$ denotes the normal distribution
(ii) $\theta = softmax(\delta)$.

Both generative processes for LDA and ETM are similar. The differences are the following:

- Instead of drawing topic from a *Dirichlet* prior, the ETM draws the parameter θ for topic distribution from a *Logistic Normal* distribution [38].
- In the last step of word generation, instead of using *Categorical* distribution, ETM model uses *softmax* calculated by the matrix multiplication of word embedding vectors (ρ) and topic embedding vectors (α_z).

The inference in ETM is made by a Variational Inference algorithm (also called amortized inference) introduced by [39]. We further give details of the inference in the following chapters. However, the crucial part is that the method allows us to apply a gradient descent algorithm. Hence, we can train our topic model in an end-to-end fashion.

We use the topic modeling approach of ETM as the backbone of our proposal. However, there are differences. First and foremost is the Neural Network estimator for the mean and covariance of the untransformed topic proportion's reparameterized

form. Here, instead of a fully-connected neural network, we leverage a graph neural network architecture, specifically, a graph attention network called TopicGNN. Further details are in the following chapters.

4. APPROACH

4.1. Model Overview

Our model has two main components: a Graph Neural Network architecture that learns entity embedding vectors and a topic model architecture that learns topic vectors in the same space of entities. We propose combining two approaches to bring a new aspect into the topic analysis. The model takes the tweet set,$\mathcal{T}$, as input and provides topic distributions of each tweet,$\mathbf{t}$.

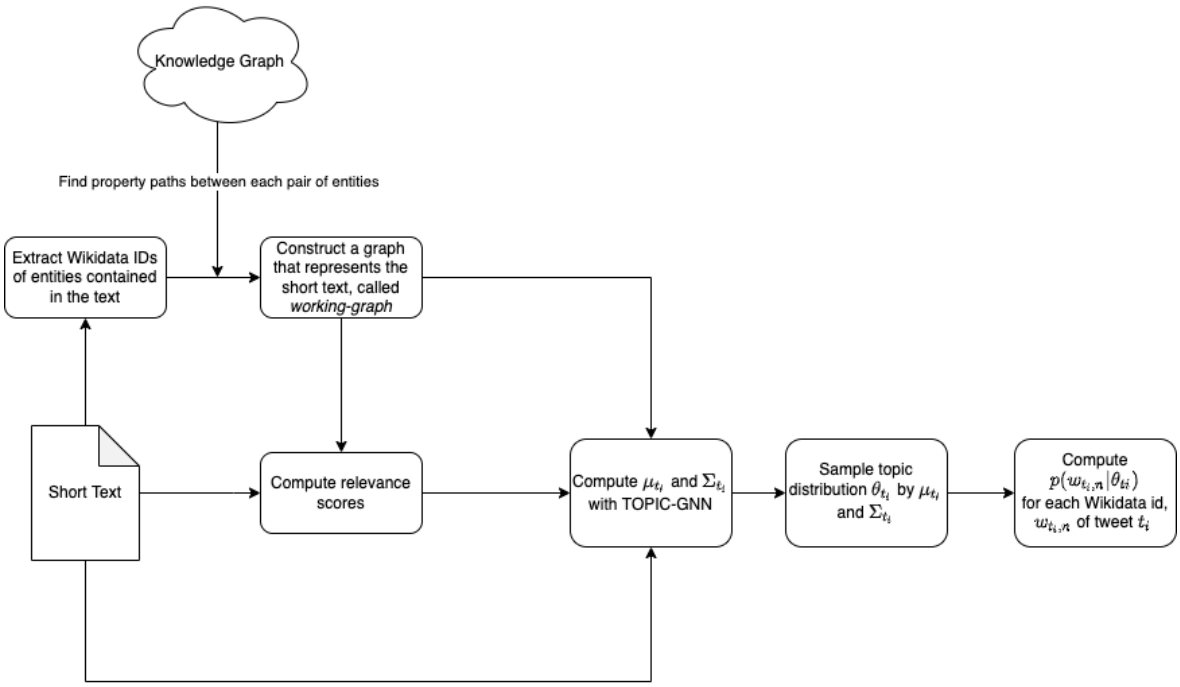

Figure 4.1. Overview of our proposed model where μ_{t_i} and Σ_{t_i} are parameters used for sampling δ_{t_i} which is the parameter for the distribution of θ_{t_i}.

Figure 4.1 depicts the structure of our model pipeline and how it combines two different models. In our topic modeling, topic proportions (θ_{t_i}) are sampled from a Logistic Normal ($\mathcal{LN}$) distribution that transforms a standard Gaussian random

variable to the simplex. The parameters μ_{t_i} and Σ_{t_i} correspond to the mean and variance of that Gaussian random variable. Those variables (μ_{t_i} and Σ_{t_i}) are estimated by our GNN architecture. Moreover, probability of each entity is estimated by using the topic proportions θ_{t_i}.

Mainly, our architecture has three stages. First, it creates a graph representation of a short text, *tweet*. In this stage, we first extract entities, then fetch all pairwise property paths between each pair of extracted entities to a predefined path length. We use two techniques while fetching property paths: creating a local sub-knowledge graph from Wikidata and searching paths on the sub-knowledge graph, and leveraging the Wikidata SPARQL Query Engine. We will explain the details of each technique later.

4.1.1. Embedded Topic Model

In Embedded Topic Model [6], the authors proposed a new topic modeling method based on word embedding vectors. The general topic modeling approach introduced by [1] inspired them. Instead of using words as pieces of text, in Embedded Topic Model [6], they used word vector embedding vectors. The Continues Bag of Words (CBoW) type of word embeddings introduced in [19] impressed the Embedded Topic Model [6] model so that it uses a similar likelihood function for the word embeddings.

In this project, we leveraged the idea of using embedding vectors to learn topics. The main difference between LDA [1] and Embedded Topic Model [6], the latter learns topic embedding vectors instead of learning words contained in topics. We constructed a topic model that learns topic and entity embedding vectors by following a similar approach. Our model's distinction is that instead of using words as a source of documents, our model considers entities linked to a knowledge graph.

4.1.2. QAGNN

The QAGNN [9] solves a question-answering problem with the help of a knowledge graph, $\mathcal{KG}$. The trained model first extracts the entities from a question using both entity embedding vectors and context embedding vectors (extracted from pure text), and it predicts possible answers. In our model, we used the idea of merging context embedding vector and entity vectors and learning representation of entities by a Graph Neural Network based model.

4.1.3. Our Model

We propose a new topic model. Contrary to traditionally adopted Latent Dirichlet Allocation [1], we modeled our topics distributions over entities instead of words. We aimed to better understand short texts (tweets) and extract related topics from them. As Figure 4.1 indicates, we integrated the knowledge graph's entity embedding vectors produced by our Graph Neural Network module feed into the Embedded Topic Model [6] structure. By having entity embedding vectors instead of word embedding vectors, we aimed to explain better and better understand tweets because of the short context of tweets. Since entities have relations in an external knowledge graph, we expect that this will make our model discover more knowledge outside of the text.

First, we prepared our data set by processing the set of tweets. We first extracted all of the annotations from each tweet and collected them into a common database. Given the set of annotations, we linked all possible annotations with the entities from Wikidata. Then we have extended our entity by using Wikidata relationships. In addition, we introduce an extra relationship between the entities extracted from the text, namely co-occurrence.

Secondly, our Graph Neural Network [21] model is based on Graph Attention Mechanism [8]. An entity embedding vector is updated based on its neighbors in this framework. As described in the background section, nodes propagated the information

through a message-passing approach based on attention. This architecture aims to learn a good representation for the node that takes information from its neighbors. This kind of information propagation is very natural because, as we think as human beings, every related entity would help us understand the target entity better. Moreover, integrating the context node that would correspond to the textual information helps not to miss any information from the text.

4.2. Definitions & Notations

4.2.1. Tweet

This section defines the concepts and types we use in our model. We first introduce the *basic tweet*. Let **b** be a BasicTweet, **b** represented as

$$\mathbf{b} =< id, context\_annotations, author\_id, text, entities, created\_at > . \tag{4.1}$$

Additionally, let **et** be a ExtendedTweet, **et** represented as

$$\mathbf{et} = [\mathbf{b}, clean\_text, tagme\_annotations, wikidata\_responses, wikidata\_ids]. \tag{4.2}$$

4.2.2. Knowledge Graph

A *context annotation* in our knowledge graph construction process is from TWITTERcontext annotations. Additionally, when we query the SPARQL endpoint of Wikidata API, we extend the set of context annotations by *subclassOf* and *instanceOf* relationships. The set of *content annotations* contains the annotations which are not context annotations.

4.2.3. Working Graph

A *working graph* is a graph we process in the training procedure for each tweet in our data set. The working graph is input for our *GAT* architecture. This graph has two types of nodes, namely context nodes and entity nodes. Context node represents tweet's text, whereas entity nodes represent the entities extracted from the knowledge graph. In constructing a working graph, we first fetch property paths between two entities and extend the graph by these paths.

The other concept we introduced while constructing working graphs is the *relevance score*. Since all the entities extracted from the knowledge graph in the working graph are not related to a tweet, we calculate a score to measure how the context node and entity nodes are related.

4.3. Data Set Processing

4.3.1. Tweet Pre-processing

We clean tweet **t**'s text by removing URLs, mentions, reserved words for TWITTER, and emoticons. Algorithm 4.2 explains the process of text cleaning in detail.

Require tweet-text
Remove followings from tweet-text:
- URLs
- mentions
- reserved words
- emojis
- smiley

Figure 4.2. Pre-process algorithm.

4.3.2. TAGME Annotations

After cleaning each tweet, $\mathbf{t}$, we send a request to TAGME API [31] to link entities. We store all TAGME responses for each tweet $\mathbf{t}$.

The TAGME API supports two fields to adjust the confidence of the entities, namely link probability that measures the reliability of that substring as a significant mention, and rho value which measures the goodness of annotation within the other entity annotations. In our process, we used link probability as 0.1 and rho value as 0.3. We selected these values by observing extracted annotations sampled tweet texts for different fields values.

4.3.3. Linking Annotations with Wikidata Entities

Before constructing our knowledge graph, $\mathcal{KG}$, we first linked the extracted entities with the entities in a public Knowledge Graph. In this work, we have used Wikidata as our source of knowledge. We first needed to match all possible annotations with Wikidata entities in the process. We achieved this goal by using Wikipedia API. The Figure 4.3 shows an example request.

```
GET https://www.wikidata.org/w/api.php
?action=wbsearchentities
&search=Jennifer
&format=json
&language=en
&strictlanguage=1
&limit=10
&type=item
```

Figure 4.3. Example entity linking query for Wikidata API.

4.3.4. Normalize Annotations

After collecting annotations from the different sources, we have normalized the annotations into one set. By normalizing, we refer to extracting text representations of the different annotations by eliminating the other fields included in an annotation item. The final annotations set consists of unique set of annotations. The Algorithm 4.4 shows how we normalize the annotations collected from various sources.

Require context-annotations, entity-annotations, tagme-annotations, ρ, link-probability

Fetch titles from context-annotations

Fetch titles from entity-annotations

Filter tagme-annotations with ρ and link-probability

Fetch titles from tagme-annotations

Combine all annotations and create annotation set

Figure 4.4. Annotation Normalization Algorithm.

4.3.5. Extended Tweets

As a first step, we collect data from the TWITTER API v2 [28] via a query determined beforehand. The data collection time may vary depending on the application or characteristics of the data set. Figure 4.5 shows the overall process of merging different annotations in a diagram.

As described in Figure 4.5, we then stored the collected data. After that, we extracted three different types of annotations.

- Context Annotations sourced by TWITTER API
- Entity Annotations sourced by TWITTER API
- Entitiy Annotations sourced by TAGME API [32]

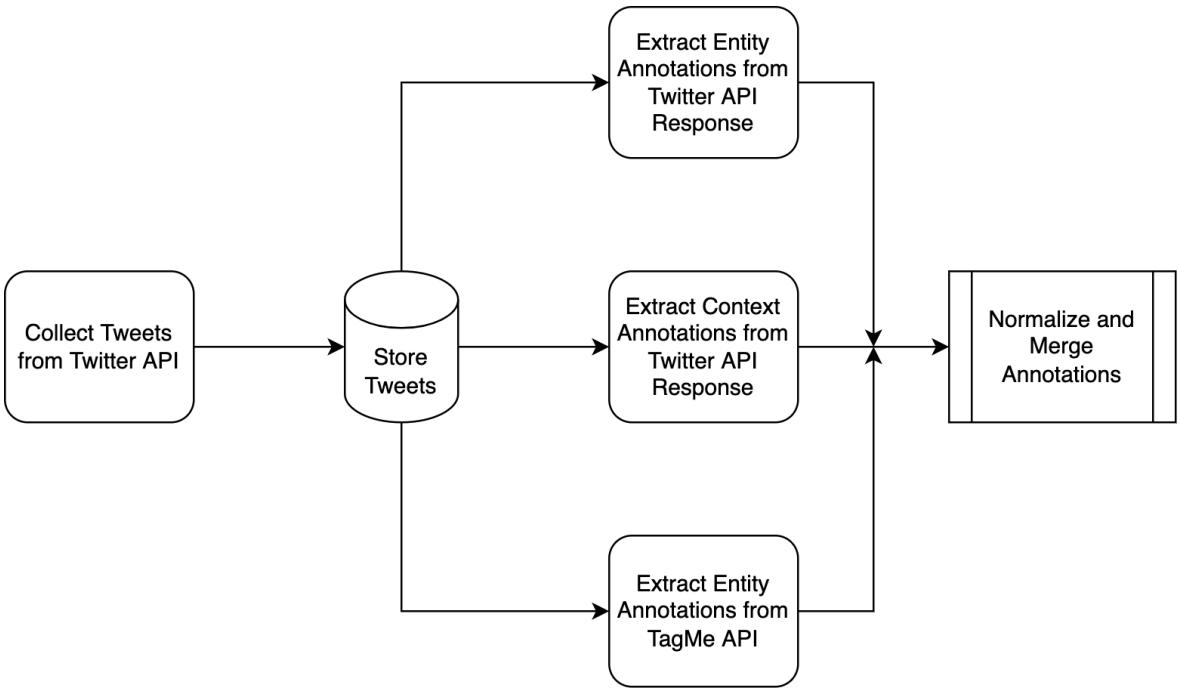

Figure 4.5. This diagram explains how our entity extraction process works.

In our database, we keep $\mathbf{et}_i$ for each $\mathbf{b}_i$. The Algorithm 4.6 explains clearly how we construct an extended tweet $\mathbf{et}_i$

Require $\mathbf{b}_i,\ i = 1, 2, \ldots, N$

tweet-text $\Leftarrow$ Extract text from $\mathbf{b}_i$

cleaned-text $\Leftarrow$ Pre-proces tweet-text

entity-annotations $\Leftarrow$ Extract entity annotations from $\mathbf{b}_i$

tagme-annotations $\Leftarrow$ Fetch TAGME annotations from database

context-annotations $\Leftarrow$ Extract context annotations from $\mathbf{b}_i$

annotations $\Leftarrow$ Normalize entity-annotations, tagme-annotations, context-annotations

wikidata-responses $\Leftarrow$ Get responses of annotations from Wikidata API by Listing 4.3

wikidata-ids $\Leftarrow$ Extract Wikidata unique ids from wikidata-responses

Construct and store extended tweet with $\mathbf{b}_i$, cleaned-text, entity-annotations, tagme-annotations, context-annotations, wikidata-responses, wikidata-ids

Figure 4.6. Algorithm that explains how an extended tweet, $\mathbf{et}_i$ is constructed.

4.3.6. Property Paths

We used two different techniques to fetch property paths between tweet entities. Firstly, we created a local knowledge graph or sub-knowledge graph, a subgraph extracted from a broader knowledge graph, Wikidata. Secondly, we used directly the SPARQL Query Engine of Wikidata to get all property paths. In both cases, we looked up a predefined path length for a property path.

4.3.6.1. Local Knowledge Graph. Due to the limitation of computational power and limiting the knowledge graph's scope, we have decided to take a sub-graph of the giant knowledge graph, Wikidata. So, instead of leveraging the whole Wikidata as or source of knowledge, we constructed our knowledge graph $\mathcal{KG}$. Algorithm 4.7 describes all

the steps we followed to construct our knowledge graph, kg, for each data set.

Our knowledge graph is not tweet-specific. Hence, we construct it at the beginning of our work. Then, we used it in various places in our algorithm.

Require $\mathbf{e}_i, i = 1, 2, \ldots, N$
knowledge-graph ← Initialize an empty graph
context-entities ← Initialize an empty set
content-entities ← Initialize an empty
for $i = 1$ to N **do**
  if $\mathbf{e}_i$ included in context annotations **then**
    Add $\mathbf{e}_i$ to context-entities
  end if
  if $\mathbf{e}_i$ included in entitiy annotations or tagme annotations **then**
    Add $\mathbf{e}_i$ to content-entities
  end if
end for
for $k = 1$ to K where K is the length of context-entities **do**
  Get *subclassOf* and *instanceOf* relationships of context-entities[k]
  Extend the $\mathcal{KG}$
end for
for $k = 1$ to K where K is the length of content-entities **do**
  Get all object relationships of content-entities[k]
  Extend the $\mathcal{KG}$
  for $j = 1$ to J where J is the number of relationships **do**
    if r_j is type of *subclassOf* or *instanceOf* **then**
      Add related entity object to context-entities
    else
      Add related entity object to content-entities
    end if
  end for
end for

Figure 4.7. For each data set, we construct a new knowledge graph, $\mathcal{KG}$ by this procedure.

for $k = 1$ to K where K is the length of newly added context-entities **do**
  Get *subclassOf* and *instanceOf* relationships of context-entities[k]
  Extend the $\mathcal{KG}$
end for
for $k = 1$ to K where K is the length of the newly added content-entities **do**
  Get all object relationships of content-entities[k]
  Extend the $\mathcal{KG}$
end for

Figure 4.7. For each data set, we construct a new knowledge graph, $\mathcal{KG}$ by this procedure. (cont.)

The idea is that, basically, we first categorize all the annotations we extracted from each tweet into two categories, namely context-entities and content-entities. The list of context-entities is initialized by the context annotations we have in $\mathbf{et}_i$ in our set of extended tweets, and other annotations initialize the list of content-entities in $\mathbf{et}_i$. Afterward, we apply two different procedures for context and content entities accordingly. For context entities, we only fetch two types of relationships from the Wikidata. This decision is because we empirically observed that context-entities tend to be more abstract. Thus, they cover more concepts, entities and this results in a bunch of entities that are not very informative to be included in our knowledge graph, $\mathcal{KG}$. On the other hand, we observed that content-entities are more refined and informative than context-entities. Therefore we fetch all the object relationships of content entities.

Postprocessing. After constructing the knowledge graph, we applied it to filter on entities. Firstly, we applied a filter that filters all instances or subclasses of creative work. However, that filter was very generic. Hence, it eliminated some valuable entities. After careful selections, we ended up following classes to filter:

(i) Music Release Type
(ii) Film
(iii) Television Program
(iv) Book

We used the Wikidata SPARQL query engine to determine if an entity is an instance of unwanted classes. So, the postprocessing filtering is introduced by *MINUS* operation of SPARQL.

A $\mathcal{KG}$ in our study is a graph consisting of multiple types of edges. Nodes in a knowledge graph represent entities, and the typed edges between entities represent relationships between entities. In general, knowledge graphs are represented with triples. In this project, we also represented a knowledge graph as a triple consisting (s, p, o) where s and o are nodes, and p is the typed edge between them.

We constructed our $\mathcal{KG}$ via a greater one. After linking our annotations with Wikidata entities, we fetched the object properties of the entities. Object property of an entity means that the property's value is again another entity rather than a value. In the scope of this work, we only fetched the direct properties of an entity and constructed our knowledge graph by using these triples. To fetch the direct properties of the entities, we have used the SPARQL query service of Wikidata.

The above sample query fetches all direct object properties of the entities *Q36159*, and it filters some certain type of relationships. The filtered relationships are

- described by the source
- topic's main category
- on focus list of Wikimedia
- topic's main template
- topic's main Wikimedia portal
- all the relationships, instances of Wikidata property

```
SELECT ?s ?sLabel ?p ?pLabel ?o ?oLabel WHERE {
  ?s ?pp ?o.
  ?p wikibase:directClaim ?pp.
  ?p wikibase:propertyType wikibase:WikibaseItem.

  MINUS {
    ?s wdt:P31+/wdt:P279* wd:Q106043376 .
  }

  MINUS {
    ?p wdt:P31 wd:Q18667213 .
  }

  FILTER (
  ?pp NOT IN (wdt:P1343,wdt:P910,wdt:P5008,wdt:P1424,
     wdt:P1151)
  )

  VALUES ?s {
    wd:Q36159
  }

  SERVICE wikibase:label { bd:serviceParam wikibase:
     language "[AUTO_LANGUAGE],en". }
}
```

Figure 4.8. Example SPARQL query for getting object properties.

As described in the postprocessing section, we also eliminate certain entities. We included this elimination in the SPARQL query. The SPARQL query engine handled the postprocessing procedure more efficiently.

We also decided to have such a long query after experimenting on different small queries. Having more than one query for eliminating unwanted relations and unwanted entities takes much more time than having such a big query.

```
SELECT ?source ?p1 ?o1 ?p2 ?target WHERE {
  VALUES ?source {
    wd:Q36159 wd:Q2447626 wd:Q155223
  }
  VALUES ?target {
    wd:Q36159 wd:Q2447626 wd:Q155223
  }
  FILTER(?source != ?target)
  {
    hint:Query hint:optimizer "None".
    ?source ?p1 ?target.
    ?pp1 wikibase:directClaim ?p1.
  }
  UNION
  {
    hint:Query hint:optimizer "None".
    ?source ?p1 ?o1.
    ?o1 ?p2 ?target.
    ?pp1 wikibase:directClaim ?p1.
    ?pp2 wikibase:directClaim ?p2.
  }
}
```

Figure 4.9. SPARQL Query to get paths between entities.

4.3.6.2. SPARQL Query Engine. In this case, we leveraged SPARQL directly to get all the property paths. We prepared a query for a tweet that fetches all the paths as triples. After that, we processed those triples and constructed paths.

Figure 4.9 shows our example, SPARQL Query, for a tweet. This query search triples between source and target entities. Since we do not filter any property or entity, it takes all Wikidata under consideration, unlike the previous approach.

4.3.7. Working Graph

We construct $\mathbf{WG}_i$ for each extended tweet $\mathbf{et}_i$ as described in Figure 4.10.

Require: $\mathbf{et}_i, i = 1, 2, \ldots N$, $\mathcal{KG}$

wikidata_ids $\Leftarrow$ Extract wikidata_ids from $\mathbf{et}_i$

text $\Leftarrow$ Extract cleaned text from $\mathbf{et}_i$

$\mathbf{WG}_i \Leftarrow$ Initialize with fetched property paths

$\mathbf{WG}_i \Leftarrow$ Extend $\mathbf{WG}_i$ with co-occurrence relationships

relevance-scores $\Leftarrow$ Calculate relevance scores $\mathbf{WG}_i$

Save $\mathbf{WG}_i$ along with relevance-scores

Figure 4.10. Steps to construct working graph for each tweet, $\mathbf{t}_i$.

Figure 4.11 is the detailed procedure for fetching property paths from the local knowledge graph. For each *wikidata id*, w_id_{ij} in the set *wikidata_ids* of $\mathbf{et}_i$ we search for a *path* between w_id_{ij} and w_id_{ik} where $j \neq k$ in the knowledge graph, $\mathcal{KG}$, we constructed before. If there is a path between w_idij and w_idik, we add the path to the paths. Similarly, Algorithm 4.12 depicts how we fetch paths directly from Wikidata. Figure 4.9 shows an example query. In same way, we prepare a query for the set of entities for each tweet.

Require: wikidata_ids, $\mathcal{KG}$

paths $\Leftarrow$ []

cutoff $\Leftarrow$ 2

working-graph $\Leftarrow$ Initialize an empty graph

combinations $\Leftarrow$ Compute combinations of wikidata_ids

for $i = 0$ to I where I is the length of combinations **do**

wikidata_id-1, wikidata_id-2 $\Leftarrow$ Get wikidata_ids from combinations-i

if There is a path on $\mathcal{KG}$ between wikidata_id-1 and wikidata_id by distance *cutoff* at most **then**

Add path to paths

end if

end for

return paths

Figure 4.11. Get property paths from the local knowledge graph for given set of entities.

Require: wikidata_ids

cutoff $\Leftarrow$ 2

working-graph $\Leftarrow$ Initialize an empty graph

SPARQL Query $\Leftarrow$ Prepare property path query for wikidata_ids

paths $\Leftarrow$ Fetch paths from Wikidata Query API

return paths

Figure 4.12. Get property paths from directly Wikidata for given set of entities.

Before the relevance score calculation, we introduce a new kind of relationship to our working graphs. This relationship is called co-occurrence. We explicitly introduce this relationship to address the issue of not having any direct connection between the tweet entities. This relationship is symmetric and added between each entity pair where both are mentioned in the tweet's text.

Moreover, we also compute one more thing within the process, namely *relevance score*. Our graph attention network uses this score to give a relative importance to the nodes of $\mathbf{WG}_i$. We declared this process in Algorithm 4.13.

Require: $\mathbf{et}_i$

relevance-scores $\Leftarrow$ Initialize an empty array

tweet-text $\Leftarrow$ Extract clean text from $\mathbf{et}_i$

for $i = 0$ to I where I is the number of nodes in subgraph **do**

input-to-language-model $\Leftarrow$ concatenate tweet-text and node n_i'title

Compute relevance score by running language model with input-to-language-model

Append relevance score to relevance-scores

end for

Compute relevance score by only tweet-text and append to relevance-scores

Figure 4.13. Calculate relevance score for each node in working graph.

4.4. Topic GNN

Our model consists of two main structures. The first one is responsible for topic modeling, and the second main structure produces entity embedding vectors and integrates them into topic modeling architecture. In this sense, Topic-GNN corresponds to the part responsible for embedding vectors producer.

In Topic-GNN, we first prepare data set for the language model. In our work, we leveraged Roberta [40] as our language model. We used Roberta [40] because, in terms of performance, it outperforms most of the transformer-based models that have available pre-trained options. Moreover, the Roberta model's training data set is considerably big. It has 1000% more data compared to BERT [41] model So, we tried to achieve the best coverage of words by using the Roberta model.

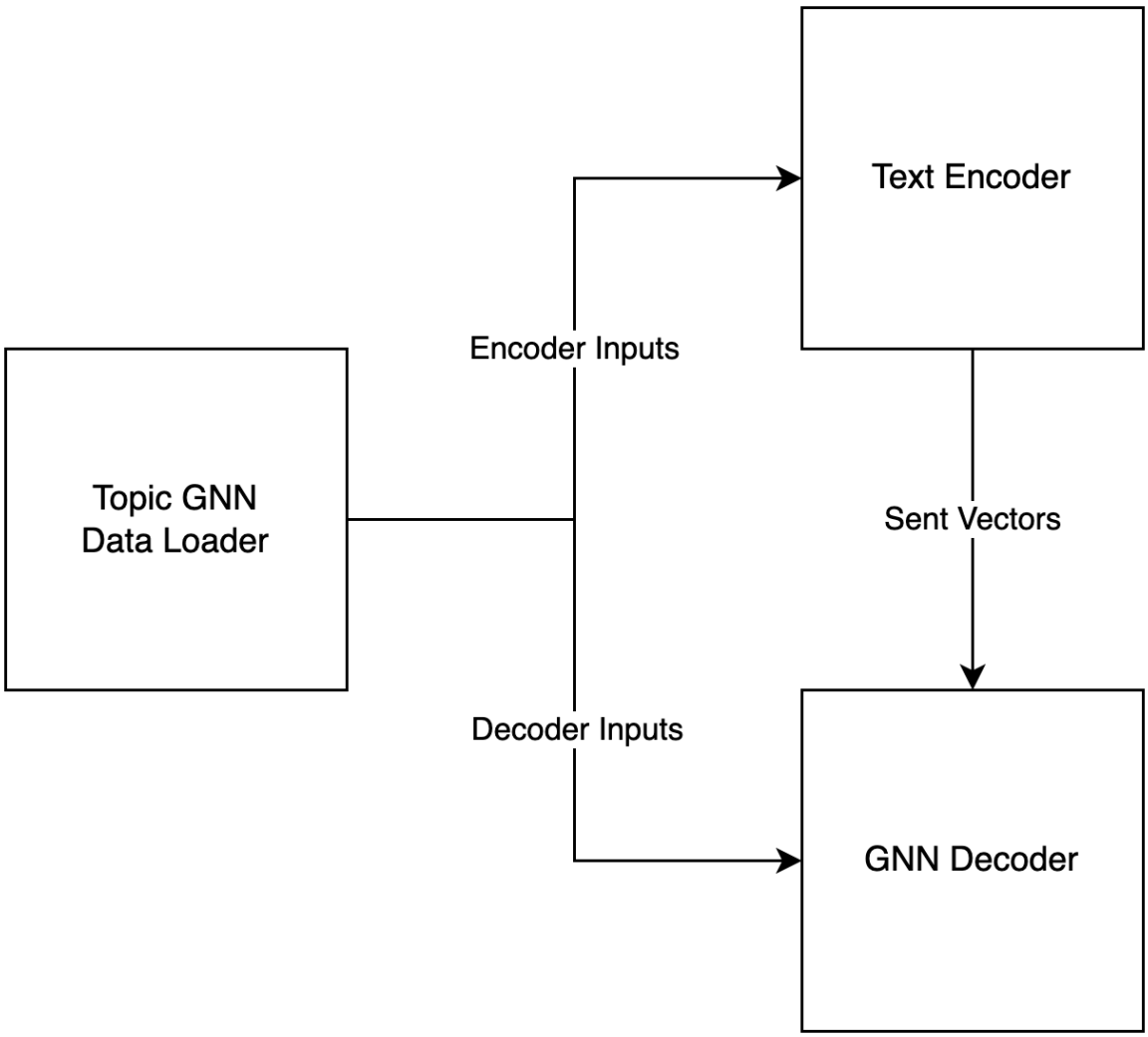

Figure 4.14. This figure depicts Topic-GNN architecture in detail.

There are two types of inputs in Topic-GNN. The first kind of input is encoder-input for the encoder based on the language model. The second kind of input is decoder input which is for the Graph Neural Network [21] based decoder. Topic-GNN-Data-Loader prepares both types of input. We discussed the details of the model parts in the implementation Chapter 5.

The GNN Decoder uses the message passing architecture designed for Graph Attention Transformer Networks [8]. Furthermore, it takes a graph as an input. In that graph, the node features are the embedding vectors for each node. The GAT has

a similar idea as the popular attention mechanism [27]. So, the attention mechanism is constructed by node neighbors instead of words in the same sentence as done in a well-known attention mechanism. After processing the input graph for multiple layers, the network produces the value used to sample the δ parameter used in topic modeling architecture. Moreover, the embedding layer of Topic-GNN is also propagated to the topic modeling structure.

4.5. Topic Modeling with Entity Embedding Vectors

Topic modeling is a complex problem, and in the literature, various works are trying to extract the topics from natural texts. The original LDA [1] architecture is one of the most common approaches used for this problem. Even though the method has shortcomings, we have seen that the model performs very well in the literature. Hence, this model influenced its successors. The LDA [1] architecture has also impacted the central architecture of our model.

We followed the similar idea of using the embedding vectors [6] of topics and the pieces that form the topics. Nevertheless, we considered topics as a distribution over entities instead of words. In this way, we aimed to increase the expressiveness of tweets by utilizing external knowledge. A tweet has a limited number of characters, and this limit is small compared to the newspaper articles or the scientific publications which the LDA and the original ETM model used as training data sets. Therefore, instead of using words included in a tweet, we decided to use entities contained in a tweet to represent itself.

The Figure 4.15 shows how our topic modeling algorithm works in detail. The main difference from Embedded Topic Model [6] is that instead of using multi layer perceptron layer to calculate μ and Σ, we leveraged Topic-GNN.

for Iteration $i = 1, 2, 3, \ldots$ **do**
  Compute $\beta_k = softmax(\rho^\top \alpha_k)$ for each topic k
  Take a batch of extended tweets
  for $j = 1$ to J where J is the number of tweets in the batch **do**
    $x_d \Leftarrow$ Get prepared GNN inputs for $\mathbf{et}_j$
    Compute $\mu_d = \text{TOPIC-GNN}(x_d; \nu_\mu)$
    Compute $\Sigma_d = \text{TOPIC-GNN}(x_d; \nu_\Sigma)$
    Sample θ_d with $\theta_d = softmax(\delta_d)$, $\delta_d = \mu_d + \Sigma_d^{\frac{1}{2}} \cdot \epsilon_d$, and $\epsilon_d \sim \mathcal{N}(0, I)$
    for each entity in extended tweet $\mathbf{et}_j$ **do**
      Compute $p(w_{dn}|\theta_d, \rho, \alpha) = \theta^\top \beta_{\cdot, w_{dn}}$
    end for
  end for
  Estimate Evidence Lower Bound (ELBO)
  Take gradients of the ELBO via backpropagation
  Update model parameters
end for

Figure 4.15. Topic modeling training process for each step where ρ is the entity embedding vectors and α is the topic embedding vectors.

4.6. Inference and Estimation

For a given data set of extended tweets,$\mathbf{et}_i$, we followed a similar inference path as in [6]. As it has been done in previous topic modeling algorithms, we optimized the marginal likelihood. In the equation

$$\mathcal{L}(\alpha, \rho) = \sum_{\mathbf{et}_i} \log p(\mathbf{et}_i|\alpha, \rho), \tag{4.3}$$

α corresponds to the embedding vectors of topics, whereas ρ is the embedding vectors of entities. Unfortunately, this marginal likelihood is not tractable. This is because the integral,

$$p(\mathbf{et}_i|\alpha,\rho) = \int p(\delta) \prod_{e\in\mathbf{et}_i} p(e|\delta,\alpha,\rho)d\delta \tag{4.4}$$

$$p(e|\delta,\alpha,\rho) = \theta\beta_e \tag{4.5}$$

$$\beta_e = softmax(\rho^\top\alpha)|_e \tag{4.6}$$

is intractable. To estimate the integral, similar to [6], we have used *Variational Inference* to estimate $p(\delta)$. We first selected a variational distribution $q(\delta;\mathbf{et}_i,\nu)$ where the proposed distribution $q(.)$ depends on the tweet we work on and the shared inference network parameters, ν. The shared inference network indicates the graph neural network, Topic-GNN, we proposed. The Topic-GNN takes the working graph prepared for the extended tweets as input and outputs the mean and covariance of the Gaussian distribution $q(\delta;\mathbf{et}_i,\nu)$.

So, using the distribution $q(\delta;\mathbf{et}_i,\nu)$ supports us to bound the marginal likelihood. This bound is called *evidence lower bound*, ELBO. The equation of the ELBO is

$$\mathcal{L}(\alpha,\rho,\nu) = \sum_{\mathbf{et}_i}\sum_{e\in\mathbf{et}_i} \mathbf{E}_q\left[\log p(e|\delta,\rho,\alpha)\right] \tag{4.7}$$

$$- \sum_{\mathbf{et}_i} KL(q(\delta;\mathbf{et}_i,\nu)||p(\delta)). \tag{4.8}$$

Again, since the expectation is intractable in the ELBO formula, we approximated it by a Monte Carlo approximation.

The ELBO becomes

$$\tilde{\mathcal{L}} = \frac{1}{S} \sum_{\mathbf{et}_i} \sum_{e \in \mathbf{et}_i} \sum_{s=1}^{S} \log p(e|\delta^s, \rho, \alpha) \tag{4.9}$$

$$- \sum_{\mathbf{et}_i} KL(q(\delta; \mathbf{et}_i, \nu)||p(\delta)) \tag{4.10}$$

$$\delta^s \sim \; q(\delta; \mathbf{et}_i, \nu). \tag{4.11}$$

And δ^s is sampled from $q(\delta; \mathbf{et}_i, \nu)$ as

$$\epsilon^s \sim \mathcal{N}(0, I) \text{ and } \delta^s = \mu + \Sigma^{\frac{1}{2}} \cdot \epsilon^s. \tag{4.12}$$

This sampled ELBO function is optimized by ADAM [42] optimization algorithm.

5. IMPLEMENTATION

5.1. Data Processing

This section will explain the implementation details of the data processing part.

5.1.1. Data Collection

In this project, we leveraged the Twitter API v2 as our source of the tweet data sets. To collect data in vast amounts, we implemented python scripts that are using the searchtweets python package. The searchtweets library is the official v2 tool of the Twitter API. For each data set, we followed the following path.

- Determine the best query for the topic
- Choose to include the retweets or not
- Start the script to fetch tweets from the current time
- Save the collected tweets to our database

As the database, we used the MongoDB version of 5.0.2. We have selected Mongo DB [43] because of the scalability and the speed of reading access. When the data is structured with key-value pairs and accessing a particular feature is more important than a whole data point, a NoSQL database is preferable over SQL. Again, in [44], other advantages of NoSql databases over SQL are explained, such as speed and compatibility with current frameworks. Considering these advantages, we have selected to use Mongo DB.

5.1.2. Entity Extraction

After collecting the tweets, we extracted the entities in a tweet. As the algorithm 4.6 describes, we extracted three different types of annotations from each tweet

and extended the tweet with them. Then, we linked the annotations with entities in Wikidata by leveraging Wikidata's search API. Although we have mentioned the usage of TAGME API, it is worth mentioning it here again. We have used the endpoint provided by the TAGME Project [32]. This favorable tool helps us to identify various annotations from short texts. Moreover, it accomplishes effectively. To manage this tool, we first registered into the system. After having the access token to its API, we adopted it in the *requests* from *Python standard library* to make the appropriate request.

5.1.3. Property Path Extraction

Since we have used two different techniques here, we separated the implementation logic.

5.1.3.1. Local Knowledge Graph Construction. As discussed earlier, we extracted our Knowledge Graph, $\mathcal{KG}$, to use in our model for each data set. This decision is made due to resource limitations because it is infeasible to include whole Wikidata as our knowledge source. We extracted our entities of interest by using the SPARQL query engine [45]. Therefore, we moved the computation of relationships and post-processing part to Wikidata's side. It helped us to boost our processing procedure. For querying the engine, we again used the requests library from Python. An example query and overall process are in Subsection 4.3.6.1.

We have performed our calculations with the NetworkX [46] Python package to represent our Knowledge Graph, $\mathcal{KG}$. This package provides a vast amount of functionalities, and it helps to compute different statistics over the graph. It has an excellent API structure and is designed specifically for graphs. We used the *MultiDiGraph* type of the package to represent $\mathcal{KG}$. This type of graph corresponds to a *multi-edged directional graph*. While constructing our knowledge graph, $\mathcal{KG}$, we decided to eliminate some of the entities of specific types. In the implementation, we leveraged a compound SPARQL query described earlier.

5.1.3.2. Fetching Paths Directly from Wikidata. This system leverages the Query Inference Engine of Wikidata. For performance improvements, we used Python's built-in *asynio* library in addition to *aiohttp* library, which is a third-party library that helped to make asynchronous HTTP calls to the corresponding endpoint.

5.1.4. Working Graph Extraction

The Topic-GNN model requires a graph for each document in the data set. In our scenario, a document corresponds to a tweet. So, we prepared a working graph for each tweet included in our data set. For achieving this goal, we used the NetworkX [46] library. The library provides functions; has_edge, all_simple_edge_paths. We used the all_simple_edge_paths function to fetch all the paths between two entities, nodes in the knowledge graph. We set a threshold for the paths we are looking for because finding all paths have two shortcomings. First, computationally it makes the process much harder. Second, having long distant paths will include various entities along the way, and although we use relevance score to indicate node importance, we determined that could cause significant expressiveness issues for the tweet representation. Hence, we set the longest path threshold to 2. We selected two by our practical experience. We tested for various thresholds and observed that the number 2 is the most efficient choice without performance loss.

5.1.4.1. Relevance Score Calculation. We calculate the relevance score based on an extended tweet's clean text and entity labels. We concatenate each entity in an extended tweet with the tweet's text. Then, we give resulted in text-entity pairs into the language model. For this calculation, we have used the transformers library.

5.1.5. Language Model Inputs

In addition to the working graph, the Topic-GNN model requires another type of input: the tweet's text. The model uses this text to represent the context node. In the preparation step, we computed the inputs given to the language model in the training

process. To be able to do that, we leveraged the transformers library [47]. The library provides vast transformer-based [27] pre-trained models and data processing tools. We used a tokenizer for the Roberta Model given by the library. This tokenizer takes text as input and outputs token id for tokens included in a sentence.

5.2. Model Training

This section focuses on implementation details of the language model, Graph Attention Transformer based Topic-GNN, and Entity Embedded Topic Model (EETM). In general, we have leveraged PyTorch [34] package as our deep learning framework. Moreover, we included additional packages written on PyTorch,

5.2.1. Language Model

We leveraged the transformers python package [47] for pre-trained language models as described before. The package has functions to download and use the transformer model. The Transformers supports two different deep learning frameworks, namely PyTorch and TensorFlow. Since we selected PyTorch [34] as our deep learning framework, we used it as the transformers backend.

5.2.2. Graph Attention Transformer

We used the Graph Attention Transformer architecture as described before. The structure is the same as [9]. Implementation of message-passing and GAT layer are based on the torch-geometric library [35].

5.2.3. Embedded Topic Model

As in the GAT part, we have also used the PyTorch [34] library in the topic modeling part. In training, we trained our model with a mini-batch gradient descent algorithm. As our optimizer, we have selected the Adam [42] as our optimization

algorithm. ADAM is an adaptive learner that allows the learning rate automatically by the length of computed gradients and the momentum.

5.2.4. Data Loader

To feed our model, we prepared a data loader that transforms the prepared data set to proper input for language model, GAT, and EETM. The data loader takes database credentials and configurations as parameters, and iteratively it loads data for training. At first, we had a high memory consumption issue, but with our data loader's chunked database loading implementation, we have overcome this issue. It also supports prediction time data loading. In that case, we load the required data efficiently by pruning unnecessary training-time inputs.

6. EXPERIMENTS AND RESULTS

We have experimented with our approach on various data sets. Afterward, we evaluated the results both qualitatively and quantitatively.

6.1. Experiments Environment

In the beginning, we have conducted our experiments on a CPU-based machine that has an Intel Core i9 CPU, 2,3 GHz, and eight cores with 32 GB RAM capacity. In this setup, training took a considerably long time, almost 2 minutes per step. Later on, we moved our data into a Cloud-Based Virtual Machine with GPU support with an Intel Broadwell CPU, four cores, 26 GB RAM capacity, and one NVIDIA Tesla K80 GPU with 11 GB memory. The GPU support increased the training performance incredibly, almost twenty times less.

6.2. Data Sets

When the volume of data increases, it is infeasible to store whole data in memory. To overcome this issue, we have created a database based on *MongoDB*. In every level of computation, we stored our data in the database. We read data from the database with chunks in pre-processing, training, and prediction times. Therefore, we kept our memory consumption low at a time.

We have collected, pre-processed the following data sets:

- Black Lives Matter
- Covid 19
- January 6
- Populist Leaders
- NBA

For each data set, we first collected and stored the responses from TWITTER API v2. We have leveraged *Academic* license provided by the TWITTER. Due to computational concerns, four of our five collected datasets are considerably mid-size data sets. However, we experimented on one more big data set to confirm our proposal in scale.

Table 6.1. Number of data points (size) in each data set and the query used to fetch data from TWITTER API.

Data Set	Size	Query	Time Interval
BLM	124274	#BLM OR #BlackLivesMatter	2021-08-02 - 2021-01-06
January 6	60402	#January6 OR #CapitolRiots OR #insurrection	2021-01-01 - 2022-01-07
Covid 19	177523	#sars-cov-2 OR #covid19	2021-12-23 - 2022-01-06
Populist Leaders	12259	from:realDonaldTrump OR from:BorisJohnson OR from:narendramodi	2018-12-09 - 2022-01-10
NBA	292573	#NBA #basketball	2008-11-07 - 2021-09-22

Table 6.1 shows the final volume of each data set and the corresponding query we used to fetch the data set. We collected unique tweets for each data set. In our work, we call a tweet is unique if it is not a reply to an existing tweet or not a retweet.

For each data set, we extracted a *knowledge graph* from Wikidata. Table 6.2 depicts the size of each knowledge graph.

Table 6.2. Details of *Knowledge Graphs* constructed for each data set.

Data Set	# of nodes	# of edges	# of edge types
BLM	465524	1641438	1125
January 6	300508	945001	1033
Covid 19	565266	1883699	1133
Populist Leaders	84872	182711	776
NBA	623453	2333805	1131

Table 6.3. Comparison of two path extraction methods.

Approach	Local Knowledge Graph				
Data Set	Populist Leaders	January 6	BLM	Covid 19	NBA
# of Tweets $\geq$ 1 node	14469	73442	133041	242930	292573
# of Tweets $\geq$ 1 link	12076	55936	123639	75838	292569
Avg. # of nodes	13.85	7.4	10.85	6.72	19.87
Avg. # of edges	50.11	31.75	54.35	37.11	112.37
Approach	Wikidata SPARQL API				
Data Set	Populist Leaders	January 6	BLM	Covid 19	NBA
# of Tweets $\geq$ 1 node	14469	73452	133420	242932	292573
# of Tweets $\geq$ 1 link	12093	56091	123638	176901	292569
Avg. # of nodes	21.2	8.97	13.49	8.93	37.49
Avg. # of edges	74.01	36.75	63.65	39.73	146.24

6.2.1. Black Lives Matter

As the reaction against racism increases around the globe, this phenomenon has become mentioned on social media more frequently. Hence, we chose this topic to collect as an example data set. We experimented on analyzing which themes are discussed on TWITTER under this sensitive topic.

We have collected 255159 tweets from the API. This query term matches the tweets that include the #BlackLivesMatter hashtag or #BLM hashtag. There are 238850 unique tweets in the data set. 134561 tweets have context annotations fetched from the TWITTER API among the unique tweets. Afterward, we also filtered out the data points that had no entity.

6.2.2. Covid 19

We also collected tweets related to the unprecedented pandemic which took effect all around the globe. First, we have collected 242935 tweets, and similarly, we filtered some portion of it.

6.2.3. January 6

On social media, sensational occasions have a significant impact. So, we included insurrection-related tweets in our experiments. This data set includes 111893 unique tweets. Among them, 73985 tweets have context annotations. We have leveraged those who have context annotations in training.

6.2.4. Populist Leaders

To analyze the political domain, we conducted an experiment on the tweets of three considerably populist leaders who are Donald Trump, Narendra Modi, Boris Johnson. We have collected all in English tweets from these three leaders. This is done by providing the usernames of each three leaders. Table 6.4 depicts the number of tweets for each leader.

Table 6.4. The number of Tweets for *Populist Leaders* data set.

Leader	# of tweets
@realDonaldTrump	14786
@BorisJohnson	4162
@narendramodi	2169

6.2.5. NBA

In the sports domain we collected tweets related to the National Basketball Association (NBA) which is a league that is popular worldwide and generates significant amounts of tweets. The query used to fetch tweets is: #NBA #basketball which fetches tweets that have #NBA and #basketball hashtags.

6.3. Experiments

Similar to LDA our approach requires setting the number of topics (K). We trained models for K= 5, 10, 15, and 20 to determine a reasonable values. We trained

models for the different k=5,10,15,20 where k is the number of topics to determine the best value. The training process took about one day for each data set and k pair. Our

Table 6.5. The values of the hyperparameters (*Learning Rate* (LR), *Batch Size* (BS), *Gradient Clipping* (GC), *Weight Decay* (WD)) used to train our models and the final loss.

Data Set	Number of topics																			
	$k = 5$					$k = 10$					$k = 15$					$k = 20$				
	LR	BS	GC	WD	Loss	LR	BS	GC	WD	Loss	LR	BS	GC	WD	Loss	LR	BS	GC	WD	Loss
BLM	0.001	32	0.0	e^{-7}	111.54	0.001	32	0.0	e^{-7}	98.24	0.005	64	0.8	e^{-6}	99.37	0.002	64	1.0	e^{-6}	112.69
January 6	0.01	32	0.0	e^{-7}	74.57	0.002	32	0.0	e^{-7}	72.15	0.001	64	1.2	e^{-6}	75.29	0.001	64	1.0	e^{-6}	85.49
Covid 19	0.005	32	0.0	e^{-7}	79.09	0.001	32	0.8	e^{-7}	73.11	0.001	64	1.0	e^{-6}	83.06	0.001	64	1.0	e^{-6}	79.84
Populist L.	0.005	32	0.0	e^{-7}	81.98	0.005	32	0.0	e^{-7}	83.04	0.005	64	0.8	e^{-6}	91.9	0.001	64	1.0	e^{-6}	80.14
NBA	0.001	64	1.0	e^{-6}	153.27	0.001	64	1.0	e^{-6}	139.38	0.005	64	1.0	e^{-6}	178.23	0.005	64	1.0	e^{-6}	141.7

model has an embedding size of 300 for topics and entities. In GNN, we used the graph attention network with two attention heads.

```
ETM(
  (t_drop): Dropout(p=0.5, inplace=False)
  (theta_act): ReLU()
  (q_theta): LMTOPICGNN(
    (encoder): TextEncoder(
      (module): RobertaModel
    )
    (decoder): TOPICGNN(
      (concept_emb): CustomizedEmbedding(
        (emb): Embedding(Vocab Size, 300)
        (cpt_transform): Linear(in_features=300, out_features=100, bias=True)
        (activation): GELU()
      )
      (svec2nvec): Linear(in_features=768, out_features=100, bias=True)
      (activation): GELU()
      (gnn): TOPICGNNMessagePassing(
        (emb_node_type): Linear(in_features=3, out_features=50, bias=True)
        (emb_score): Linear(in_features=50, out_features=50, bias=True)
        (edge_encoder): Sequential(
          (0): Linear(in_features=1132, out_features=100, bias=True)
          (1): BatchNorm1d(100, eps=1e-05, momentum=0.1, affine=True, track_running_stats=
              True)
          (2): ReLU()
          (3): Linear(in_features=100, out_features=100, bias=True)
        )
        (gnn_layers): ModuleList(
            (0): GATConvE()
            (1): GATConvE()
            (2): GATConvE()
            (3): GATConvE()
            (4): GATConvE()
          )
```

Figure 6.1. Created model architecture for Entity Embedded Topic Modeling.

```
                (Vh): Linear(in_features=100, out_features=100, bias=True)
                (Vx): Linear(in_features=100, out_features=100, bias=True)
                (activation): GELU()
                (dropout): Dropout(p=0.2, inplace=False)
              )
            (pooler): MultiheadAttPoolLayer(
                (w_qs): Linear(in_features=768, out_features=100, bias=True)
                (w_ks): Linear(in_features=100, out_features=100, bias=True)
                (w_vs): Linear(in_features=100, out_features=100, bias=True)
                (attention): MatrixVectorScaledDotProductAttention(
                (dropout): Dropout(p=0.1, inplace=False)
                (softmax): Softmax(dim=1)
              )
            (dropout): Dropout(p=0.1, inplace=False)
          )
        (fc): MLP(
          (layers): Sequential(
            (0-Linear): Linear(in_features=968, out_features=1600, bias=True)
            (0-Dropout): Dropout(p=0.2, inplace=False)
            (0-LayerNorm): LayerNorm((1600,), eps=1e-05, elementwise_affine=True)
            (0-gelu): GELU()
            (1-Linear): Linear(in_features=1600, out_features=800, bias=True)
          )
        )
        (dropout_e): Dropout(p=0.2, inplace=False)
        (dropout_fc): Dropout(p=0.2, inplace=False)
        )
      )
        (alphas): Linear(in_features=300, out_features=20, bias=False)
        (mu_q_theta): Linear(in_features=800, out_features=20, bias=True)
        (logsigma_q_theta): Linear(in_features=800, out_features=20, bias=True)
    )
```

Figure 6.2. Created model architecture for Entity Embedded Topic Modeling (cont).

6.4. Evaluation

Although many of the classical topic models take the number of topics higher than fifty, we selected this number, k, as 5, 10, 15, 20 because, in our data sets, we focus on specific areas such as sports, politics, and sensational events. Hence, we do not expect those datasets include more than 20 topics.

6.5. Quantitative Results

We evaluated our model's extracted topics using topic coherence and topic diversity scores, where the topic coherence score is the C_{UMass} score for the top 10 words and topic diversity based on the top 25 words. Moreover, we compared topic coherence and topic diversity with LDA [1]. Typical LDA takes inputs as Bag of Words representation of documents. However, we prepared LDA's inputs from our prepared knowledge graph entities to compare our results. For instance, given a tweet,

#donaldtrump bears moral responsibility for the #capitolriots, writes. but the senates job is to defend the constitution, not find some way, any way, to inflict a punishment.

We take the following list of Wikidata ids as BoW representation of a tweet:

"Q22686", "Q164746", "Q66096", "Q164746", "Q2570643"

Table 6.6. Topic Coherence (TC) and Topic Diversity (TD) results of LDA for each data set, number of topics, k

Data Set	Topic Coherence				Topic Diversity			
	k=5	k=10	k=15	k=20	k=5	k=10	k=15	k=20
BLM	−4.558	−4.820	−7.294	−8.534	0.904	0.892	0.931	0.914
January 6	−3.637	−3.584	−7.965	−10.875	0.880	0.892	0.901	0.926
Covid 19	−6.236	−8.043	−9.424	−9.827	0.944	0.940	0.952	0.940
Populist Leaders	−7.162	−9.001	−11.834	−12.534	0.880	0.856	0.842	0.834
NBA	−2.509	−3.714	−3.199	−2.884	0.860	0.852	0.835	0.740

Table 6.7. Topic Coherence (TC) and Topic Diversity (TD) results for each data set, number of topics, k pair without co-occurrence relationship and property paths are constructed by local knowledge graph method

Data Set	Topic Coherence				Topic Diversity			
	k=5	k=10	k=15	k=20	k=5	k=10	k=15	k=20
BLM	−4.949	−5.026	−6.775	−12.754	1.000	1.000	0.994	0.944
January 6	−1.767	−2.959	−6.966	−9.393	1.000	0.996	0.994	0.972
Covid 19	−4.097	−5.905	−11.144	−10.961	0.992	0.996	0.906	0.790
Populist Leaders	−1.900	−6.688	−7.769	−11.114	1.000	0.988	0.874	0.952
NBA	−2.437	−7.086	−7.212	−6.043	1.000	1.000	0.997	0.992

Table 6.8. Topic Coherence (TC) and Topic Diversity (TD) results for each data set, number of topics, k pair with co-occurrence relationship and property paths are constructed by local knowledge graph method

Data Set	Topic Coherence				Topic Diversity			
	k=5	k=10	k=15	k=20	k=5	k=10	k=15	k=20
BLM	−5.519	−5.371	−5.618	−6.901	1.000	0.988	0.986	0.956
January 6	−3.770	−4.230	−6.680	−7.220	0.980	0.940	0.900	0.870
Covid 19	−5.708	−8.439	−8.037	−9.659	0.992	0.896	0.885	0.846
Populist Leaders	−7.488	−5.378	−10.510	−9.820	0.690	0.460	0.620	0.720
NBA	−2.440	−3.441	−4.707	−5.809	1.000	0.992	0.986	0.980

Table 6.9. Topic Coherence (TC) and Topic Diversity (TD) results for each data set, number of topics, k pair with co-occurrence relationship and property paths are constructed by querying SPARQL API of Wikidata

Data Set	Topic Coherence				Topic Diversity			
	k=5	k=10	k=15	k=20	k=5	k=10	k=15	k=20
BLM	−5.840	−5.460	−6.621	−7.151	1.000	0.980	0.965	0.912
January 6	−2.240	−3.270	−4.700	−5.470	1.000	0.920	0.950	0.930
Covid 19	−6.890	−6.630	−8.150	−8.216	0.960	0.992	0.890	0.910
Populist Leaders	−1.840	−3.330	−3.150	−5.070	1.000	1.000	0.960	0.950
NBA	−1.830	−2.820	−4.300	−5.127	1.000	1.000	1.000	0.980

Table 6.7 shows results from the run without the co-occurrence relationship. With this setup, our model poorly performs compared to LDA, see Table 6.6. This provides strong evidence that the LDA model reflects co-occurrence relationships and to perform

equally, and they need to be explicitly defined. Moreover, we have used the local-knowledge-graph strategy here. Contrast, Table 6.8 shows results after we introduced the co-occurrence relationship into our pipeline. Significantly, it improves our results for the majority of data sets. Moreover, we provided the property paths by exploiting the SPARQL API, which improved our results further, see Table 6.9. As explained earlier, the main difference is that we do not filter out any relationship or entity types in local-knowledge-graph-based property path calculation. We observed that our model learns considerably diverse topics compared to the classical LDA approach. Furthermore, the coherence scores increase when the number of topics increases.

6.6. Qualitative Results

We analyzed our results in different ways. First, we evaluated our learned embedding vectors of entities and topics. As we expected, learned entity embedding vectors are close to the topic embedding vectors. Figure 6.3 depicts the entity and topic embedding vectors together. In our model, the embedding vector has a dimension of 300, but we have used t-SNE to reduce the size for display purposes. We concluded that our model learns coherent entity and topic embedding vectors from these results.

Secondly, we experimented with the entity embedding vectors in a downstream task after our primary motivation. We represented a tweet as a combination of its entities' vectors. Figure 6.3 shows the tweets representations and colored clusters. For display purposes, here, we used t-SNE to reduce the dimensionality. These results show the power of entity embedding vectors learned during topic modeling.

Table 6.10 shows the top ten topics generated for the NBA dataset. A manual inspection reveals that many of the topic elements are reasonable an what might be expected. The topic elements include famous basketball players such as LeBron James (one of the all-time best players in NBA), highly influential basketball teams such as Cleveland Cavaliers and Miami Heat for his career as well as the division they belong to (Altantic Division). Topic 8 is related to Boston Celtics, a very famous

American professional basketball team that competes in the NBA (National Basketball Association) league. One of the elements of this topic is Bill Russel who is historically one of the most famous players of Boston Celtics. Topic 3 is related to entertainment and the NBA. We observed Kobe Bryant and 2019 FIBA Basketball World Cup entities. Kobe Bryant was global ambassador of the tournament.

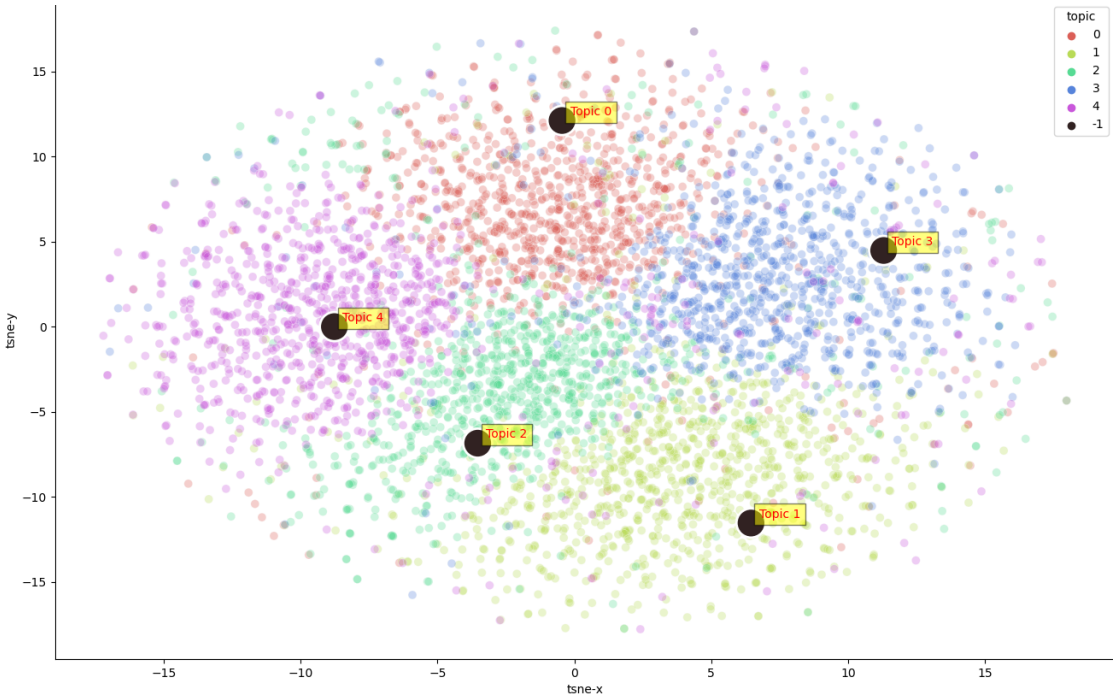

Figure 6.3. Visualization of the top 1000 entity embedding vectors for each topic for the BLM dataset (k=5) using t-SNE

Table 6.10 shows the results for ten topics extracted for NBA data set. Other results for each k included in Appendix A.

Moreover, we analyzed each topic by calculating co-occurrences of each entity pair in top words. Co-occurrence graph, Figure 6.5, depicts the top fifteen entities in Topic 4 of the NBA data set, trained with the number of topics ten. As the results indicate, our model captures more co-occurred entities as representatives of topics. So, we concluded that although our model performance is worse than or comparable to the LDA in a quantitative manner, qualitatively, it shows promising results.

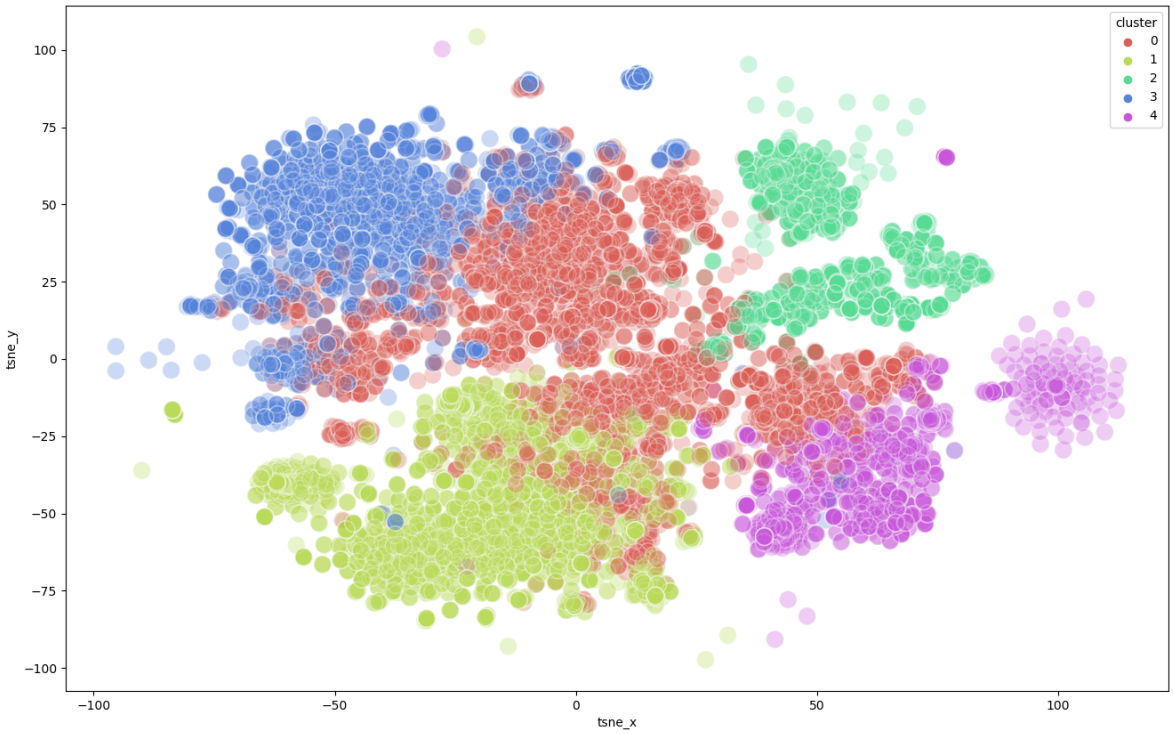

Figure 6.4. Visualization of the tweets clustered based on entity embeddings with K-Means for the Populist Leaders dataset using t-SNE

Table 6.10. Top 10 words for each topic trained over all data sets for $k = 10$ for NBA data set

Topic 0	Olympic sport, ball game, National Hockey League, Canada, villain, A systematic review on gender-specific suicide mortality in medical doctors., Yuri Mikhailovich Kopylov, Wakanim, Elliott Roosevelt, Mamie Hui
Topic 1	team sport, New Orleans Pelicans, Nick Nurse, Dream 11, Madison Square Garden, Reuben Kemper, Rai Radio 1, Phosphodiesterase 3A, Tony A. Garofano, Montereau-Fault-Yonne
Topic 2	basketball player, All-NBA Team, NBA All-Star Game Kobe Bryant Most Valuable Player Award, NBA Rookie of the Year Award, shooting guard, Golden State Warriors, Brooklyn Nets, Western Conference, NBA TV, NBA Summer League
Topic 3	entertainment, Major League Baseball, ESPN, Denver Nuggets, Portland Trail Blazers, 2019 FIBA Basketball World Cup, Kobe Bryant, FanDuel, fashion, 2020–21 NBA season
Topic 4	basketball team, NBA Most Valuable Player Award, Miami Heat, power forward, NBA All-Defensive Team, Best NBA Player ESPY Award, Cleveland Cavaliers, Atlantic Division, LeBron James, Chicago Bulls
Topic 5	Basketball, Eastern Time Zone, Minnesota Timberwolves, LeBron, National Collegiate Athletic Association, Cryptocurrencies and Zero Mode Wave guides: An unclouded path to a more contiguous Cannabis sativa L. genome assembly, Chinook, Calgary Metropolitan Region, Vrané nad Vltavou, Mayor of Wy-dit-Joli-Village
Topic 6	National Basketball Association, baseball, NBA Most Improved Player Award, North Carolina Tar Heels men's basketball, Tołwin, physical fitness, Oscar Robertson Trophy, San Antonio Spurs, TD Garden, wrestling at the 2018 Commonwealth Games
Topic 7	men's basketball, NCAA Division I men's basketball, Los Angeles Lakers, small forward, NBA G League, NCAA Division I, Commissioner of the NBA, Oklahoma City Thunder, 2018–19 NBA season, Toronto Raptors
Topic 8	sport, point guard, Bill Russell NBA Finals Most Valuable Player Award, gambling, Boston Celtics, retail, Ohio Mr. Basketball, NBA Finals, Central Division, military exercise
Topic 9	basketball, NBA All-Rookie Team, National Football League, Phoenix Suns, NBA Draft Lottery, Los Angeles Clippers, Dallas Mavericks, Basketball Association of America, John R. Wooden Award, eBay

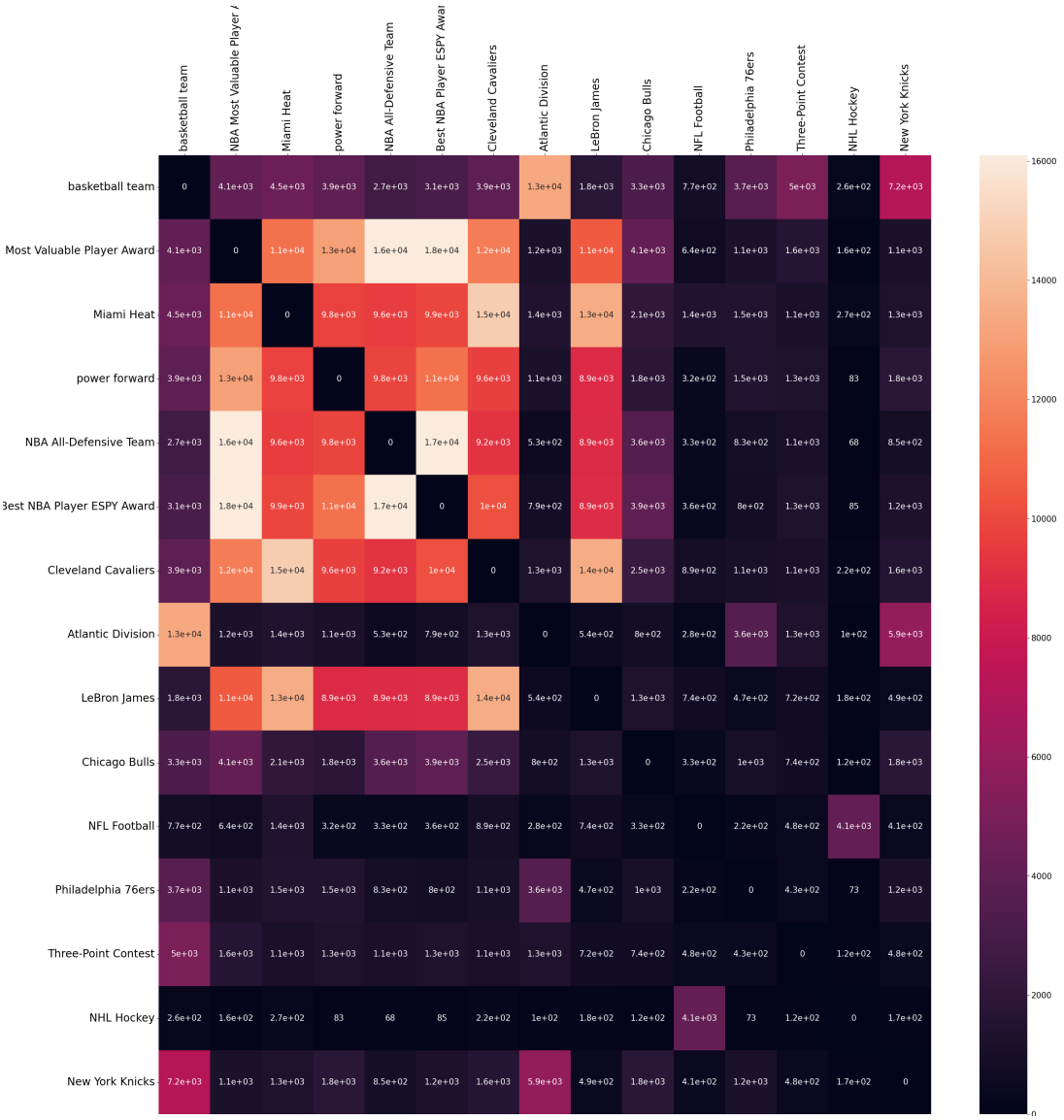

Figure 6.5. Co-occurrence plot of top fifteen entities of Topic 4 for NBA Data Set

7. DISCUSSION & FUTURE WORK

This thesis proposes a new topic modeling method based on entities extracted from documents and linked to an external knowledge graph. Moreover, the model learns distributed representations of entities and topics. To the best of our knowledge, this work is the first one that combines the knowledge graph embedding and topic modeling.

We tested our model on various data sets of tweets collected from TWITTER in evaluation. Our results indicate that the model learns interpretable and expressive topic and entity embedding vectors. When we utilized these embedding vectors to express tweets, we observed that these representations could benefit downstream tasks. This representative power is a significant contribution because it is difficult to decompose a set of tweets, and our vectors streamline this task.

Moreover, our quantitative results show that the model shows worse or comparable results regarding the topic coherence metric. This issue could originate from various reasons. In our algorithm, extracted knowledge graphs for a data set play a significant role because we formed working graphs which are inputs to the GAT network established on the knowledge graphs. Furthermore, we constructed the knowledge graphs based on tweets entities. Focusing on that part, we observed some shortcomings, such as in the extraction process, the mechanism might miss essential entities. For instance, in our deep analysis of the results, when we looked at BLM tweets, it missed the annotation of KKK, a white supremacist terrorist, and hate group, due to the low rho value returned from TagMe API.

The main problem regarding the knowledge graph is that our source of knowledge, Wikidata, only provides recent relationships. For instance, when we fetch the relationships of the United States of America, we only get recent properties such as the current president. Although the information is correct, the information of previous

presidents is missing. Hence, we observed that some of the expected paths between two entities are missing, e.g., Donald Trump and Joe Biden.

When we analyze the extracted entities through topics, country names come forward. It means that the location information dominates within the data sets. Although it could be desirable in some cases, it creates a noise mainly. Hence, we concluded that location information requires careful handling.

Additionally, we filtered many entities from the knowledge graphs as described earlier. However, we still observed various questionable entities in extracted topics, e.g., unrelated locations and people names. Despite the questionable quantitative results, we observed that our model catches important entities in topics for a data set. Moreover, topic diversity shows better results, indicating that the model could infer more diverse topics in terms of entities.

Although it brings questionable entities, utilizing a knowledge graph and entities has various advantages. First, entities are at a higher level than words in semantics. For instance, in a sentence that includes a neural network, a word-by-word approach would consider neural and network separately, whereas the entity approach would consider the neural network as one entity. Second, relationships of entities bring crucial information to the model regarding knowledge discovery. As a concrete example, consider the sentence "Last night, the game between Cavaliers and Warriors was fantastic". A word-based system would need many samples to relate two basketball teams, Cleveland Cavaliers and Golden State Warriors. However, from the knowledge graph, our model constructs this relation immediately.

7.1. Error Analysis

Extracting topics is highly challenging when the sort texts are concerned. On the road, we encountered many difficulties. The major problem is the lack of context. Our proposal aims to overcome this issue.

We observed that the country names dominate some topics for every data set. It indicates the location information requires careful handling. Even though it might be the desired result for some data sets, in our data sets, such as National Basketball Association, country names are not an expected topic. Secondly, when we investigate the top entities of each topic, there are shortcomings in our entity linking process. There are entities such as "Cryptocurrencies and Zero Mode Wave guides: An unclouded path to a more contiguous Cannabis sativa L. genome assembly" and "Wellness and health omics linked to the environment: the WHOLE approach to personalized medicine." Those entities from Wikidata correspond to specific scientific articles, but the intended entities were "Cryptocurrencies" and "Wellness and health." Since we link annotations from the text by Wikidata Search API, the results can deviate from what we wanted. Another issue regarding extracted topics is misleading annotations coming from TAGME API. The API extracts unexpected annotations from the text. For instance, it extracts "severe acute respiratory syndrome" from the text that contains "sarscov."

7.2. Future Work

We have achieved many refinements on the constructed knowledge graphs. However, our results and deeper analysis indicate that there is still room for improvement. Primarily, we figured that location and person entities dominate, and further refinements on these as a possible next step will bring significant improvements.

Moreover, we have utilized entity embedding vectors in the tweet clustering task. These embedding vectors could also be utilized in a classification task of tweets. Also, one could use them in a knowledge graph completion task.

Finally, testing our model on corpora with longer texts could help to compare the model with previous works. Due to keeping the scope of this work more focused, we left these as future work.

8. CONCLUSION

Topic modeling is an exciting and complex problem, elaborated in literature for years. However, even the state-of-the-art models have shortcomings regarding the documents with short texts due to the limited context and knowledge. This thesis proposed a new topic model to overcome these shortcomings.

The goals of this work were threefold; developing a new document representation, an entity-context graph, learning distributed vector representations of these entities, and extracting interpretable topics and their distributed vector representations. Firstly, representing documents as graphs of entities and linking them into an external knowledge graph contributed to opening the door to the Linked Open Data world. Secondly, the model learns entity embedding vectors via joint learning with topics, and we showed that entity vectors have the power to represent documents discriminatively. Then, the model provides interpretable topics that help understand documents' hidden structures.

Our results showed that the proposed model produces powerful entity and topic embedding vectors; indeed, we utilized them in a downstream task to observe document expressiveness. Besides, our quantitative results showed our model's vulnerabilities stated in Chapter 7. In conclusion, our qualitative analysis showed promising results despite the quantitative results. So, with future improvements, our model would outperform its competitors in the topic modeling.

REFERENCES

1. Blei, D. M., A. Y. Ng and M. I. Jordan, "Latent Dirichlet Allocation", *Journal of Machine Learning Research*, Vol. 3, No. 4-5, pp. 993–1022, 2003.

2. Rosen-Zvi, M., T. Griffiths, M. Steyvers and P. Smyth, "The Author-Topic Model For Authors and Documents", *arXiv preprint arXiv:1207.4169*, 2012.

3. Zhu, J. and E. P. Xing, "Conditional Topic Random Fields", *Proceedings of the 27th International Conference on International Conference on Machine Learning*, pp. 1239–1246, 2010.

4. Blei, D. M. and J. D. Lafferty, "Dynamic Topic Models", *Association for Computing Machinery International Conference Proceeding Series*, Vol. 148, pp. 113–120, 2006.

5. Shen, D., C. Qin, C. Wang, Z. Dong, H. Zhu and H. Xiong, "Topic Modeling Revisited: A Document Graph-based Neural Network Perspective", *Advances in Neural Information Processing Systems*, Vol. 34, pp. 14681–14693, 2021.

6. Dieng, A. B., F. J. Ruiz and D. M. Blei, "Topic Modeling in Embedding Spaces", *Transactions of the Association for Computational Linguistics*, Vol. 8, pp. 439–453, 2020.

7. Vrandečić, D. and M. Krötzsch, "Wikidata: A Free Collaborative Knowledgebase", *Communications of the Association for Computing Machinery*, Vol. 57, No. 10, pp. 78–85, 2014.

8. Veličković, P., G. Cucurull, A. Casanova, A. Romero, P. Lio and Y. Bengio, "Graph attention networks", *arXiv preprint arXiv:1710.10903*, 2017.

9. Yasunaga, M., H. Ren, A. Bosselut, P. Liang and J. Leskovec, "Qa-gnn: Reasoning

with Language Models and Knowledge Graphs for Question Answering", *arXiv preprint arXiv:2104.06378*, 2021.

10. Petroni, F., T. Rocktäschel, P. Lewis, A. Bakhtin, Y. Wu, A. H. Miller and S. Riedel, "Language Models as Knowledge Bases?", *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pp. 2463–2473, 2020.

11. Pan, X., K. Sun, D. Yu, J. Chen, H. Ji, C. Cardie and D. Yu, "Improving Question Answering with External Knowledge", *Proceedings of the 2nd Workshop on Machine Reading for Question Answering*, pp. 27–37, 2019.

12. Ye, Z.-X., Q. Chen, W. Wang and Z.-H. Ling, "Align, Mask and Select: A Simple Method for Incorporating Commonsense Knowledge into Language Representation Models", *arXiv preprint cs.CL/1908.06725*, 2019.

13. Bosselut, A., H. Rashkin, M. Sap, C. Malaviya, A. Celikyilmaz and Y. Choi, "CoMET: Commonsense Transformers for Automatic Knowledge Graph Construction", *Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference*, pp. 4762–4779, 2020.

14. Trouillon, T., J. Welbl, S. Riedel, E. Gaussier and G. Bouchard, "Complex Embeddings for Simple Link Prediction", M. F. Balcan and K. Q. Weinberger (Editors), *Proceedings of The 33rd International Conference on Machine Learning*, Vol. 48, pp. 2071–2080, 2016.

15. Bordes, A., N. Usunier, A. Garcia-Duran, J. Weston and O. Yakhnenko, "Translating Embeddings for Modeling Multi-relational Data", *Proceedings of the 26th International Conference on Neural Information Processing Systems*, Vol. 2, pp. 1–9, 2013.

16. Wang, X., X. He, Y. Cao, M. Liu and T.-S. Chua, "KGAT: Knowledge Graph Attention Network for Recommendation", *Proceedings of The 25th Association for*

Computing Machinery International Conference on Knowledge Discovery & Data Mining, pp. 950–958, 2019.

17. Feng, Y., X. Chen, B. Y. Lin, P. Wang, J. Yan and X. Ren, "Scalable Multi-Hop Relational Reasoning for Knowledge-Aware Question Answering", *arXiv preprint arXiv:2005.00646*, 2020.

18. Zhou, D., X. Hu and R. Wang, "Neural Topic Modeling by Incorporating Document Relationship Graph", *arXiv preprint arXiv:2009.13972*, 2020.

19. Mikolov, T., K. Chen, G. Corrado and J. Dean, "Efficient Estimation of Word Representations in Vector Space", *1st International Conference on Learning Representations, Workshop Track Proceedings*, pp. 1–12, 2013.

20. Lauscher, A., P. Ruiz Fabo, F. Nanni and S. P. Ponzetto, "Entities as Topic Labels: Combining Entity Linking and Labeled LDA to Improve Topic Interpretability and Evaluability", *Italian Journal of Computational Linguistics*, Vol. 2, No. 2-2, pp. 67–87, 2016.

21. Scarselli, F., M. Gori, A. C. Tsoi, M. Hagenbuchner and G. Monfardini, "The Graph Neural Network Model", *Institute of Electrical and Electronics Engineers Transactions on Neural Networks*, Vol. 20, No. 1, pp. 61–80, 2009.

22. Wu, Z., S. Pan, F. Chen, G. Long, C. Zhang and S. Y. Philip, "A Comprehensive Survey on Graph Neural Networks", *Institute of Electrical and Electronics Engineers Transactions on Neural Networks and Learning Systems*, Vol. 32, No. 1, pp. 4–24, 2020.

23. Kipf, T. N. and M. Welling, "Semi-supervised Classification with Graph Convolutional Networks", *5th International Conference on Learning Representations, Conference Track Proceedings*, pp. 1–14, 2017.

24. Dumoulin, V. and F. Visin, "A Guide to Convolution Arithmetic for Deep Learn-

ing", *arXiv preprint arXiv:1603.07285*, 2016.

25. Bahdanau, D., K. Cho and Y. Bengio, "Neural Machine Translation by Jointly Learning to Align and Translate", *arXiv preprint arXiv:1409.0473*, 2014.

26. Lin, Z., M. Feng, C. N. d. Santos, M. Yu, B. Xiang, B. Zhou and Y. Bengio, "A Structured Self-attentive Sentence Embedding", *arXiv preprint arXiv:1703.03130*, 2017.

27. Vaswani, A., N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser and I. Polosukhin, "Attention Is All You Need", *Advances in Neural Information Processing Systems*, Vol. 30, pp. 5999–6009, 2017.

28. Twitter, *Twitter API v2*, 2021, `https://developer.twitter.com/en/docs/twitter-api`, accessed in December 2021.

29. Jin, W., Y. Ma, Y. Wang, X. Liu, J. Tang, Y. Cen, J. Qiu, J. Tang, C. Shi, Y. Ye *et al.*, "Graph Representation Learning: Foundations, Methods, Applications and Systems", *Proceedings of the 27th Association for Computing Machinery Conference on Knowledge Discovery & Data Mining*, pp. 4044–4045, 2021.

30. Ehrlinger, L. and W. Wöß, "Towards A Definition of Knowledge Graphs.", *SEMANTiCS (Posters, Demos, SuCCESS)*, Vol. 48, No. 1-4, p. 2, 2016.

31. TagMe, *TagMe API*, 2010, `https://sobigdata.d4science.org/web/tagme/tagme-help`, accessed in December 2021.

32. Ferragina, P. and U. Scaiella, "Fast and Accurate Annotation of Short Texts with Wikipedia Pages", *Institute of Electrical and Electronics Engineers Software*, Vol. 29, No. 1, pp. 70–75, 2012.

33. W3C, *SPARQL 1.1 Overview*, `https://www.w3.org/TR/2013/REC-sparql11-overview- 20130321/`, accessed in December 2021.

34. Paszke, A., S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga *et al.*, "PyTorch: An Imperative Style, High-Performance Deep Learning Library", *Advances in Neural Information Processing Systems*, Vol. 32, 2019.

35. Bronstein, M. M., J. Bruna, T. Cohen and P. Veličković, "Geometric Deep learning: Grids, Groups, Graphs, Geodesics, and Gauges", *arXiv preprint arXiv:2104.13478*, 2021.

36. Wallach, H. M., I. Murray, R. Salakhutdinov and D. Mimno, "Evaluation Methods for Topic Models", *Proceedings of The 26th Annual International Conference on Machine Learning*, pp. 1105–1112, 2009.

37. Röder, M., A. Both and A. Hinneburg, "Exploring The Space of Topic Coherence Measures", *Proceedings of The Eighth Association for Computing Machinery International Conference on Web Search and Data Mining*, pp. 399–408, 2015.

38. Blei, D. M. and J. D. Lafferty, "A Correlated Topic Model of Science", *The Annals of Applied Statistics*, Vol. 1, No. 1, pp. 17–35, 2007.

39. Kingma, D. P. and M. Welling, "Auto-encoding Variational Bayes", *arXiv preprint arXiv:1312.6114*, 2013.

40. Liu, Y., M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer and V. Stoyanov, "Roberta: A Robustly Optimized BERT Pretraining Approach", *arXiv preprint arXiv:1907.11692*, 2019.

41. Kenton, J. D. M.-W. C. and L. K. Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding", *Universal Language Model Fine-tuning for Text Classification*, p. 278, 2019.

42. Kingma, D. P. and J. L. Ba, "Adam: A Method for Stochastic Optimization", *3rd International Conference on Learning Representations, Conference Track Proceed-*

ings, pp. 1–15, 2015.

43. DB, M., *Mongo DB*, 2007, `https://docs.mongodb.com/`, accessed in December 2021.

44. Stonebraker, M., "SQL Databases v. NoSQL Databases", *Communications of the Association for Computing Machinery*, Vol. 53, No. 4, p. 10–11, 2010.

45. Wikidata, *Wikidata* SPARQL *Manual*, `https://www.wikidata.org/wiki/Wikidata:SPARQL_tutorial`, accessed in December 2021.

46. Hagberg, A. A., D. A. Schult and P. J. Swart, "Exploring Network Structure, Dynamics, and Function Using NetworkX", *7th Python in Science Conference*, pp. 11–15, 2008.

47. Wolf, T., L. Debut, V. Sanh, J. Chaumond, C. Delangue, A. Moi, P. Cistac, T. Rault, R. Louf, M. Funtowicz, J. Davison, S. Shleifer, P. von Platen, C. Ma, Y. Jernite, J. Plu, C. Xu, T. L. Scao, S. Gugger, M. Drame, Q. Lhoest and A. M. Rush, "Transformers: State-of-the-Art Natural Language Processing", *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pp. 38–45, 2020.

APPENDIX A: EXTRACTED TOPICS

Table A.1. Top 10 words for each topic trained over BLM for $k = 5$.

Topic 0	George Floyd, visual arts, rap, technology, nonprofit organization, Facebook, retail, Canada, team sport, France
Topic 1	Business & Finance, United Kingdom, personal finance, United Nations, India, Turkey, 2020 United States presidential election, Australia, Democratic Party, Maillane
Topic 2	United States of America, entertainment, Christmas, President of the United States, police brutality in the United States, Poland, Republican Party, Mexico, financial services, South Africa
Topic 3	Services, COVID-19, Joe Biden, Donald Trump, Twitter, MSNBC, food, YouTube, ICC Men's T20 World Cup, Patrisse Khan-Cullors
Topic 4	Black Lives Matter, anti-fascism, cricket, sport, online, NBC News, Egypt, history, North Korea, New York City

Table A.2. Top 10 words for each topic trained over January 6 for $k = 5$.

Topic 0	George Floyd, Donald Trump Jr., Barron Trump, COVID-19, Melania Trump, Washington, D.C., Mary L. Trump, Nancy Pelosi, New York City, Time Person of the Year
Topic 1	United States of America, United States Congress, United States House of Representatives, Ivanka Trump, Donald Trump 2020 presidential campaign, Lara Trump, White House, star on Hollywood Walk of Fame, Fred Trump Jr., United States withdrawal from Iran Deal
Topic 2	Donald Trump, Eric Trump, United States Capitol, Trump–Ukraine scandal, University of Pennsylvania, Golden Raspberry Award for Worst New Star, Fordham University, 2020 Republican Party presidential primaries, Bonwit Teller Building, Elizabeth Trump Grau
Topic 3	President of the United States, Federal Bureau of Investigation, Services, second impeachment of Donald Trump, Primetime Emmy Award for Outstanding Reality-Competition Program, Democratic Party, Vanessa Trump, Jamaica Hospital, John Whitney Walter, Donald Trump 2000 presidential campaign
Topic 4	Joe Biden, 2020 United States presidential election, Republican Party, entertainment, United States Senate, inauguration of Donald Trump, Donald Trump 2016 presidential campaign, Capitol Records, Fred Trump, Tiffany Trump

Table A.3. Top 10 words for each topic trained over Covid19 for $k = 5$.

Topic 0	COVID-19 pandemic, SARSr-CoV, Business & Finance, United Kingdom, Canada, travel, General Travelling Notes, Donald Trump Jr., Taiwan, Ukraine
Topic 1	COVID-19, Israel, Serbia, Denmark, National Health Service, Georgia, transport, Warner–Lambert, Mumbai, Poland
Topic 2	COVID-19, technology, entertainment, European Union, Bangladesh, Iran, Japan, Mexico, vaccination, Delhi
Topic 3	SARS-CoV-2, United States of America, WikiProject COVID-19, New Years Eve, Australia, English, Germany, Donald Trump, State Council of the People's Republic of China, Christmas
Topic 4	India, Wellness and health omics linked to the environment: the WHOLE approach to personalized medicine., New Year's Eve, People's Republic of China, Covid Inc., data science, community health, Common medications and drugs: how they affect male fertility., Joe Biden, personal finance

Table A.4. Top 10 words for each topic trained over Populist Leaders for $k = 5$.

Topic 0	India, Boris Johnson, Democratic Party, COVID-19 pandemic, Syria, World Health Organization, Afghanistan, Belarus, Kenya, Myanmar
Topic 1	United States of America, Republican Party, 2020 United States presidential election, Lara Trump, 2016 United States presidential election, People's Republic of China, Elizabeth Christ, Russia, Saudi Arabia, France
Topic 2	Narendra Modi, United Kingdom, COVID-19, Donald Trump 2000 presidential campaign, Mexico, North Korea, Republic of Ireland, Chile, Namibia, United States Senate
Topic 3	Donald Trump, Barron Trump, Prime minister of India, Gaming Hall of Fame, Australia, Ukraine, Brazil, United Nations, United States Congress, Joe Biden
Topic 4	President of the United States, Melania Trump, Mar-a-Lago, 2020 Republican Party presidential primaries, Japan, New York City, Bangladesh, European Union, Canada, Washington, D.C.

Table A.5. Top 10 words for each topic trained over NBA for $k = 5$.

Topic 0	basketball, ball game, Olympic sport, NCAA Division I men's basketball, NBA All-Rookie Team, Pacific Division, Best NBA Player ESPY Award, 2018–19 NBA season, Michael Jordan, Services
Topic 1	Basketball, basketball player, Golden State Warriors, shooting guard, NBA All-Star Game Kobe Bryant Most Valuable Player Award, Milwaukee Bucks, Washington Wizards, center, National Collegiate Athletic Association, NBA TV
Topic 2	National Basketball Association, National Football League, small forward, power forward, NBA Most Valuable Player Award, New York Knicks, Atlantic Division, Brooklyn Nets, Bill Russell NBA Finals Most Valuable Player Award, Charlotte Hornets
Topic 3	sport, Los Angeles Lakers, point guard, Cleveland Cavaliers, Miami Heat, NBA Rookie of the Year Award, Boston Celtics, Houston Rockets, Philadelphia 76ers, Central Division
Topic 4	team sport, entertainment, men's basketball, basketball team, NFL Football, Major League Baseball, NBA G League, NBA All-Defensive Team, Toronto Raptors, New Orleans Pelicans

Table A.6. Top 10 words for each topic trained over BLM for $k = 10$.

Topic 0	Joe Biden, writing, South Africa, Martin Luther King Jr., New York City, Wellness and health omics linked to the environment: the WHOLE approach to personalized medicine., YouTube, murder of George Floyd, Olympic sport, Algeria
Topic 1	anti-fascism, cricket, Canada, sport, African Americans, retail, history, team sport, online, Washington, D.C.
Topic 2	United States of America, Christmas, President of the United States, financial services, police brutality in the United States, Republican Party, Russia, Poland, Online services., Texas A&M–Kingsville Javelinas men's basketball
Topic 3	rap, Democratic Party, food, Hungary, Greece, Australia, Germany, Taiwan, Mexico, Delhi Capitals
Topic 4	Services, Business & Finance, Twitter, Facebook, Patrisse Khan-Cullors, American football, Alicia Garza, Turkey, ICC Men's T20 World Cup, People's Republic of China
Topic 5	Black Lives Matter, Ukraine, Barack Obama, South Korea, Matamela Cyril Ramaphosa, Mali, Rachel Maddow, Netflix, Delaware Fightin' Blue Hens football, HuffPost
Topic 6	basketball, Denmark, Georgia, North Korea, Bangladesh, Belgium, NBC News, drink, physical fitness, Sidney Greenslade
Topic 7	entertainment, Basketball, George Floyd protests, Ayọ Tometi, National Football League, Italy, hip hop music, Breonna Taylor, Bernie Sanders, BBC
Topic 8	George Floyd, visual arts, personal finance, nonprofit organization, United Nations, Kamala Harris, France, investment, TikTok, Donald Trump Jr.
Topic 9	COVID-19, Donald Trump, United Kingdom, 2020 United States presidential election, MSNBC, Cryptocurrencies and Zero Mode Wave guides: An unclouded path to a more contiguous Cannabis sativa L. genome assembly, India, technology, NFL Football, Malaysia

Table A.7. Top 10 words for each topic trained over January 6 for $k = 10$.

Topic 0	President of the United States, COVID-19, Vanessa Trump, John Whitney Walter, Ellis Island Medal of Honor, Primetime Emmy Award for Outstanding Reality-Competition Program, SAG-AFTRA, Donald Trump, Trump National Golf Club, 2021 storming of the United States Capitol
Topic 1	Joe Biden, Federal Bureau of Investigation, United States Senate, Capitol Records, Services, Ted Cruz, Lauren Boebert, United States Capitol Police, Adam Kinzinger, basketball
Topic 2	United States House of Representatives, Democratic Party, first impeachment inquiry against Donald Trump, Trump: The Art of the Deal, Fred Trump Jr., Time Person of the Year, Donald Trump 2000 presidential campaign, Mar-a-Lago, Steve Bannon, Golden Raspberry Award for Worst Supporting Actor
Topic 3	Republican Party, Ivanka Trump, Fred Trump, inauguration of Donald Trump, Maryanne Trump Barry, Jared Kushner, Lara Trump, Donald Trump 2020 presidential campaign, second impeachment of Donald Trump, John G. Trump
Topic 4	Donald Trump, Melania Trump, Washington, D.C., Eric Trump, United States Capitol, Trump–Ukraine scandal, 2016 United States presidential election, University of Pennsylvania, Fordham University, Elizabeth Trump Grau
Topic 5	Donald Trump Jr., Barron Trump, Mary L. Trump, Doublespeak Award, Trump World Tower, Elizabeth Christ, The New York Times, Kamala Harris, MSNBC, United States senator
Topic 6	Donald Trump, anti-fascism, Chuck Schumer, Jill Biden, Francis Biden, Josh Hawley, penny, CBS News, Denmark, Nicolle Wallace
Topic 7	United States of America, White House, star on Hollywood Walk of Fame, United States withdrawal from Iran Deal, bibliography of Donald Trump, Gaming Hall of Fame, The Washington Post, Matt Gaetz, Mark Randall Meadows, Vice President of the United States
Topic 8	George Floyd, Tiffany Trump, Nancy Pelosi, New York City, Kevin McCarthy, Trump National Golf Club, Crippled America, Jamaica Hospital, The Wharton School, United States–Mexico–Canada Agreement
Topic 9	2020 United States presidential election, United States Congress, entertainment, Donald Trump 2016 presidential campaign, Business & Finance, Joe Biden, Facebook, International Herald Tribune, Black Lives Matter, sedition

Table A.8. Top 10 words for each topic trained over Covid 19 for $k = 10$.

Topic 0	SARSr-CoV, Business & Finance, United Kingdom, English, Christmas, Pfizer, Germany, Taiwan, G. D. Searle & Company, South Africa
Topic 1	WikiProject COVID-19, Australia, Joe Biden, pharmaceutical industry, Services, Warner–Lambert, Italy, Cryptocurrencies and Zero Mode Wave guides: An unclouded path to a more contiguous Cannabis sativa L. genome assembly, Greece, Ukraine
Topic 2	travel, General Travelling Notes, Donald Trump Jr., National Health Service, Serbia, NFL Football, Medivation, Denmark, American football, Japan
Topic 3	COVID-19, New South Wales, The New York Times, American football player, insurance, Haryanvi, Chennai, Jill Biden, Liberia, Tibetan
Topic 4	technology, New Years Eve, Donald Trump, Boris Johnson, New York City, Pfizer UK, North Korea, Brazil, Maharashtra, Doug Ford
Topic 5	New Year's Eve, Common medications and drugs: how they affect male fertility., personal finance, Israel, France, Philippines, United Nations, Narendra Modi, Wyeth, Delhi
Topic 6	SARS-CoV-2, entertainment, community health, State Council of the People's Republic of China, Bangladesh, National Football League, Malaysia, Indonesia, Mexico, Twitter
Topic 7	India, Wellness and health omics linked to the environment: the WHOLE approach to personalized medicine., Covid Inc., data science, People's Republic of China, vaccine, Russia, Ontario, Melania Trump, Turkey
Topic 8	COVID-19, Barron Trump, Thailand, London, Urdu, mental health, team sport, Ethiopia, Carole King, Ron DeSantis
Topic 9	COVID-19 pandemic, United States of America, Canada, Pakistan, Tozinameran, Hungary, New Year's Day, food, President of the United States, Scott Morrison

Table A.9. Top 10 words for each topic trained over Populist Leaders for $k = 10$.

Topic 0	President of the United States, Democratic Party, Australia, Vietnam, COVID-19 pandemic, World Health Organization, Syria, Azerbaijan, Norway, Yemen
Topic 1	Donald Trump, Texas, Kamala Harris, Adam Schiff, Somali shilling, South Asia, point bar, Federal Bureau of Investigation, Svargarohana Parva, Bosnian American
Topic 2	Melania Trump, European Union, Washington, D.C., Iraq, Canada, Russia, New York City, Gaming Hall of Fame, Israel, Japan
Topic 3	COVID-19, Barron Trump, Prime minister of India, Donald Trump 2000 presidential campaign, 2020 Republican Party presidential primaries, Brazil, New Zealand, United States Congress, Belarus, Senegal
Topic 4	Narendra Modi, White House, Bolivia, Hindi, Wellness and health omics linked to the environment: the WHOLE approach to personalized medicine., Kohn, Downing Street, Lakshmi, Hunter Biden, Jammu and Kashmir
Topic 5	India, Afghanistan, Venezuela, SARS-CoV-2, Madagascar, Croatia, personal finance, Tunisia, Maharashtra, Services
Topic 6	Republican Party, Elizabeth Christ, entertainment, South Korea, Malaysia, Nepal, Portugal, Cambodia, Sri Lanka, Argentina
Topic 7	People's Republic of China, Lara Trump, Donald Trump 2016 presidential campaign, Saudi Arabia, Philippines, Ukraine, North Korea, Republic of Ireland, Belgium, Holy See
Topic 8	United Kingdom, 2020 United States presidential election, Mexico, Indonesia, Poland, Joe Biden, Chile, Business & Finance, Brunei, National Health Service
Topic 9	United States of America, Boris Johnson, 2016 United States presidential election, Namibia, Austria, United States senator, Michael Bloomberg, Fänrik, anti-fascism, New England Colonies

Table A.10. Top 10 words for each topic trained over NBA for $k = 10$.

Topic 0	Olympic sport, ball game, National Hockey League, Canada, villain, A systematic review on gender-specific suicide mortality in medical doctors., Yuri Mikhailovich Kopylov, Wakanim, Elliott Roosevelt, Mamie Hui
Topic 1	team sport, New Orleans Pelicans, Nick Nurse, Dream 11, Madison Square Garden, Reuben Kemper, Rai Radio 1, Phosphodiesterase 3A, Tony A. Garofano, Montereau-Fault-Yonne
Topic 2	basketball player, All-NBA Team, NBA All-Star Game Kobe Bryant Most Valuable Player Award, NBA Rookie of the Year Award, shooting guard, Golden State Warriors, Brooklyn Nets, Western Conference, NBA TV, NBA Summer League
Topic 3	entertainment, Major League Baseball, ESPN, Denver Nuggets, Portland Trail Blazers, 2019 FIBA Basketball World Cup, Kobe Bryant, FanDuel, fashion, 2020–21 NBA season
Topic 4	basketball team, NBA Most Valuable Player Award, Miami Heat, power forward, NBA All-Defensive Team, Best NBA Player ESPY Award, Cleveland Cavaliers, Atlantic Division, LeBron James, Chicago Bulls
Topic 5	Basketball, Eastern Time Zone, Minnesota Timberwolves, LeBron, National Collegiate Athletic Association, Cryptocurrencies and Zero Mode Wave guides: An unclouded path to a more contiguous Cannabis sativa L. genome assembly, Chinook, Calgary Metropolitan Region, Vrané nad Vltavou, Mayor of Wy-dit-Joli-Village
Topic 6	National Basketball Association, baseball, NBA Most Improved Player Award, North Carolina Tar Heels men's basketball, Tołwin, physical fitness, Oscar Robertson Trophy, San Antonio Spurs, TD Garden, wrestling at the 2018 Commonwealth Games
Topic 7	men's basketball, NCAA Division I men's basketball, Los Angeles Lakers, small forward, NBA G League, NCAA Division I, Commissioner of the NBA, Oklahoma City Thunder, 2018–19 NBA season, Toronto Raptors
Topic 8	sport, point guard, Bill Russell NBA Finals Most Valuable Player Award, gambling, Boston Celtics, retail, Ohio Mr. Basketball, NBA Finals, Central Division, military exercise
Topic 9	basketball, NBA All-Rookie Team, National Football League, Phoenix Suns, NBA Draft Lottery, Los Angeles Clippers, Dallas Mavericks, Basketball Association of America, John R. Wooden Award, eBay

Table A.11. Top 10 words for each topic trained over BLM for $k = 15$.

Topic 0	Donald Trump, 2020 United States presidential election, NFL Football, Cryptocurrencies and Zero Mode Wave guides: An unclouded path to a more contiguous Cannabis sativa L. genome assembly, food, Olympic sport, Donald Trump Jr., White House, COVID-19 pandemic, American English
Topic 1	Black Lives Matter, sport, Matamela Cyril Ramaphosa, North Korea, NBC News, ViacomCBS, G20, HuffPost, Netflix, Brooklyn
Topic 2	rap, anti-fascism, Democratic Party, New York City, South Africa, Breonna Taylor, Denmark, hip hop music, Quinton de Kock, Delhi Capitals
Topic 3	nonprofit organization, Kamala Harris, Washington, D.C., Greece, writing, association football, Spotify, Fayetteville, Twitch, Melania Trump
Topic 4	United Kingdom, Facebook, MSNBC, African Americans, Wellness and health omics linked to the environment: the WHOLE approach to personalized medicine., Republican Party, Russia, People's Republic of China, Major League Baseball, Bangladesh
Topic 5	Joe Biden, Business & Finance, Twitter, American football, National Basketball Association, Patrisse Khan-Cullors, military exercise, ICC Men's T20 World Cup, Presidential Medal of Freedom, St. Mary's Hospital
Topic 6	United States of America, financial services, United States Congress, LGBT, travel, Terrence Floyd, Texas A&M–Kingsville Javelinas men's basketball, Diboll Unit, George Floyd, racing
Topic 7	Services, Alicia Garza, Martin Luther King Jr., ball game, pop music, murder of George Floyd, United States Senate, Pakistan, Ashley Biden, Ukraine
Topic 8	Christmas, President of the United States, National Football League, police brutality in the United States, Hungary, Mexico, Poland, K-pop, France, rapper
Topic 9	Black Lives Matter, English football league system, BLINK, Faculty of Mathematics and Geoinformation, Gwen Stacy, California, Bel-Air Village, Nicki Minaj, Loulé, Barbara Rütting
Topic 10	basketball, Canada, technology, history, cricket, team sport, George Floyd protests, Bernie Sanders, Indonesia, online
Topic 11	George Floyd, Germany, United Nations, India, Taiwan, Turkey, Afghanistan, Brazil, Australia, Joe Biden
Topic 12	entertainment, Basketball, Ayọ Tometi, Italy, YouTube, BBC, physical fitness, Instagram, Sunrisers Hyderabad, baseball
Topic 13	George Floyd, visual arts, Uniregistry, investment, Colin Kaepernick, Highveld Lions cricket team, Apple Inc., Maryanne Trump Barry, Lara Trump, Dining
Topic 14	COVID-19, personal finance, retail, fashion, drink, TikTok, Bill Spivey, Belgium, CBS, Republic of Ireland

Table A.12. Top 10 words for each topic trained over January 6 for $k = 15$.

Topic 0	United States Senate, second impeachment of Donald Trump, Donald Trump, Primetime Emmy Award for Outstanding Reality-Competition Program, John Whitney Walter, SAG-AFTRA, United States withdrawal from the Paris Agreement, MSNBC, Jamie Raskin, Tucker Carlson
Topic 1	Federal Bureau of Investigation, Bonwit Teller Building, Trump National Golf Club, Twitter, Merrick Garland, nonprofit organization, Ashley Biden, People's Republic of China, Crimean speech of President Putin, Flores Sea
Topic 2	COVID-19, Services, Vanessa Trump, Ted Cruz, first impeachment inquiry against Donald Trump, Jamaica Hospital, Ron Johnson, United States Department of Justice, Indian National Congress, contiguous United States
Topic 3	President of the United States, Jared Kushner, University of Pennsylvania, The Kew-Forest School, Ellis Island Medal of Honor, Trump National Golf Club, Elizabeth Cheney, The New York Times, Hunter Biden, César Award for Best Director
Topic 4	George Floyd, Ivanka Trump, Nancy Pelosi, Crippled America, The Wharton School, United States senator, United States representative, Chuck Schumer, The Washington Post, 1972 United States Senate election in Delaware
Topic 5	2020 United States presidential election, United States Congress, entertainment, Donald Trump 2016 presidential campaign, Joe Biden, Business & Finance, Jim Banks, Chamorro, Daniel of St. Thomas Jenifer, Jae
Topic 6	Donald Trump Jr., Washington, D.C., Eric Trump, Doublespeak Award, Ig Nobel Prize, Elizabeth Christ, Jim Jordan, Federal Government of the United States of America, Ivana Trump, Mitch McConnell
Topic 7	Melania Trump, Capitol Records, Donald Trump 2020 presidential campaign, Fordham University, Time Person of the Year, Steve Bannon, Golden Raspberry Award for Worst Supporting Actor, Matt Gaetz, Mark Randall Meadows, Facebook
Topic 8	United States of America, White House, United States withdrawal from Iran Deal, star on Hollywood Walk of Fame, bibliography of Donald Trump, deity, flag of Belarus, Rmeil, The Life of Henry the Fifth, Berkhamsted School
Topic 9	Joe Biden, Tiffany Trump, Kevin McCarthy, Trump: The Art of the Deal, Lauren Boebert, basketball, Lindsey Graham, Estates of Brittany, KLKB1, Sotira
Topic 10	Barron Trump, Mary L. Trump, United States Capitol, New York City, United States–Mexico–Canada Agreement, Trump World Tower, Elizabeth Trump Grau, Mike Pence, Kamala Harris, Beau Biden
Topic 11	Donald Trump, 2016 United States presidential election, Golden Raspberry Award for Worst New Star, antifascism, Jill Biden, Vice President of the United States, Paul Gosar, Russia, rurality, Silver tax 1816
Topic 12	Republican Party, Maryanne Trump Barry, inauguration of Donald Trump, Fred Trump, Lara Trump, John G. Trump, Trump–Ukraine scandal, 2020 Republican Party presidential primaries, Mary Anne MacLeod Trump, New York Military Academy
Topic 13	United States House of Representatives, Democratic Party, Mar-a-Lago, Donald Trump 2000 presidential campaign, Fred Trump Jr., Joseph R. Biden Sr., Christopher A. Wray, Joe Biden presidential campaign, 2008, Meta Platforms, NBC News
Topic 14	Donald Trump, 2018 United States House of Representatives elections, Fayetteville, Ashley Biden, Hip-hop theater, Redfoo discography, architectural heritage monument in Bavaria, Vratsa Municipality, DPP4, Ferrières-en-Brie

Table A.13. Top 10 words for each topic trained over Covid19 for $k = 15$.

Topic 0	COVID-19 pandemic, WikiProject COVID-19, 2002 Asian Games, William Ivins, Jr., Halifax Town A.F.C., Ifosfamide + mitoxantrone in advanced breast cancer previously treated with anthracyclines, DNA-binding transcription factor binding, Pearl River, Hercules, chess player
Topic 1	community health, Australia, Indonesia, National Football League, Services, Bangladesh, Pakistan, Germany, queen consort, Monopolies and Restrictive Trade Practices (Amendment) Act, 1991
Topic 2	COVID-19, levomepromazine, The Tragedy of King Richard the Second, demon in a work of fiction, Dhenkanal State, Bank of Hamilton, WorldPay, Sóc Trăng, Sihanouk International Airport, Giovanni Colombo
Topic 3	United States of America, data science, New Year's Day, Ontario, Turkey, regulation of amyloid-beta clearance, Kamaniskeg Lake, Asgaut Olsen Regelstad, Shaco, Janardan Mishra
Topic 4	SARS-CoV-2, entertainment, State Council of the People's Republic of China, Rashidian, Mesha, Facebook, Maya Lin, Barvinkove-Losowaja Operation, Alessandro Rigotti, Dzongkha
Topic 5	COVID-19, levomepromazine, The Tragedy of King Richard the Second, demon in a work of fiction, Dhenkanal State, Sóc Trăng, Giovanni Colombo, Bank of Hamilton, Sihanouk International Airport, Mike Cassidy
Topic 6	Covid Inc., Christmas, Germany, Russia, Taiwan, Radovan, Endothelin and nitric oxide interact to regulate stretch-induced ANP secretion., Business & Finance, Little Yarra River, lumberjack
Topic 7	Israel, United Kingdom, New Year's Eve, Heretaunga College, John J. McCall, presidential inauguration of Joe Biden, Paeonia veitchii, Dongbinhelu, Ushita, Luzon Strait
Topic 8	Business & Finance, Canada, Maharashtra, Gender differences in the decline of mortality rates of acute myocardial infarction in West Germany., Port of Lido, PU.1 regulates expression of the interleukin-7 receptor in lymphoid progenitors., Donald Trump, not threatened, São Domingos, Wörnitz
Topic 9	COVID-19, India, technology, Aldershot F.C., Georges-Charles de Heeckeren d'Anthès, Mayor of Tehran, member of the Lok Sabha, Finnegan, Nicole Johnson, canton of Tourouvre au Perche
Topic 10	Wellness and health omics linked to the environment: the WHOLE approach to personalized medicine., Tozinameran, Melania Trump, Golden Raspberry Award for Worst Supporting Actress, Richard Carapaz, Sospirolo, Réseau des sports, outline of ancient history, Smad4 dependency defines two classes of transforming growth factor beta (TGF-beta) target genes and distinguishes TGF-beta-induced epithelial-mesenchymal transition from its antiproliferative and migratory responses, unincorporated town in Nevada
Topic 11	SARSr-CoV, United Kingdom, English, Pfizer, Einheitsgemeinde Geising, THE MINISTRY, Tetracycline repressor, tetR, rather than the tetR-mammalian cell transcription factor fusion derivatives, regulates inducible gene expression in mammalian cells., Neos, Lucy Harris, Strange vision: ganglion cells as circadian photoreceptors
Topic 12	New Years Eve, travel, South Africa, The economic costs associated with physical inactivity and obesity in Canada: an update., Tsukushi controls ectodermal patterning and neural crest specification in Xenopus by direct regulation of BMP4 and X-delta-1 activity., G, Vertebrate craniofacial development: the relation between ontogenetic process and morphological outcome., regional district of British Columbia, Sierra Leone at the 2016 Summer Olympics, Le Malesherbois
Topic 13	Joe Biden, People's Republic of China, personal finance, European Union, COVID-19, Poland, Medivation, Common medications and drugs: how they affect male fertility., Artus, University of California, Berkeley Libraries
Topic 14	New Year's Eve, Common medications and drugs: how they affect male fertility., forwards, metastatic melanoma, Huatusco, MS Westerdam, People's Republic of China, Tryptase alpha/beta 1, École des Ponts ParisTech, demographics of Antarctica

Table A.14. Top 10 words for each topic trained over Populist Leaders for $k = 15$.

Topic 0	United States–Mexico–Canada Agreement, Gaming Hall of Fame, United Nations, Joe Biden, Italy, Donald Trump, United States Congress, Tajikistan, Turkmenistan, Afghanistan
Topic 1	Washington, D.C., COVID-19 pandemic, Vietnam, White House, 2020 Republican Party presidential primaries, Business & Finance, Republic of Ireland, Zambia, Iran, Brexit
Topic 2	Democratic Party, National Health Service, Donald Trump 2016 presidential campaign, Brunei, Quapaw, America/Argentina/Ushuaia, General Travelling Notes, Hermitage Museum, Sucre, Flag of Novokuznetsk
Topic 3	United States of America, 2020 United States presidential election, Indonesia, cdc-48.1, economy of Pakistan, Chemnitz, Long-Term Credit Bank of Japan, Jaishankar filmography, Austria, Nage
Topic 4	President of the United States, Melania Trump, New York City, Lara Trump, Canada, Bangladesh, European Union, Saudi Arabia, Japan, Mar-a-Lago
Topic 5	Narendra Modi, United Kingdom, Papaver rhoeas, plastic, rugby league player, United States Army infantry divisions by unit number, Pat Cipollone, Chairman of the Council of Ministers of Bosnia and Herzegovina, badminton at the 1970 British Commonwealth Games – men's doubles, Knight of Freedom Award
Topic 6	India, COVID-19 pandemic, Xia, Mitchell Whitfield, Boris Johnson, Mayor of Tampa, Florida, All India Institute of Medical Sciences, Kalyani, 1963-1964 one-year-period, administrative territorial entity of England, Borys
Topic 7	Republican Party, People's Republic of China, Elizabeth Christ, entertainment, megacity, Herakleopolis Magna, African Americans, municipality of Belgium, The Cloisters, California
Topic 8	United Kingdom, COVID-19, Narendra Modi, Brexit, India national cricket team, hypertension, Jhabua district, Ethiopian Electric Power, Matsunaga, Silvia
Topic 9	Australia, Indonesia, South Korea, Sri Lanka, World Health Organization, Cambodia, Brexit, Nepal, United States Senate, Norway
Topic 10	Boris Johnson, United States of America, United Kingdom, Fänrik, Northern English, Bisher Al-Khasawneh, Mount Herzl, Austria, Rogers, Denbighshire
Topic 11	Donald Trump, Croatia, John Witherspoon, Knight of Freedom Award, Kingdom of Rheged, Duchenne muscular dystrophy, Adi, Princess María of Greece and Denmark, Somali shilling, Michigan Data Store
Topic 12	Barron Trump, Prime minister of India, Russia, Donald Trump 2000 presidential campaign, Hungary, Brazil, Iraq, Ukraine, New Zealand, Senegal
Topic 13	2016 United States presidential election, Serbia, France, Pakistan, Mexico, Georgia, Philippines, United States of America, Syria, Austria
Topic 14	Donald Trump, John Witherspoon, Princess María of Greece and Denmark, Kingdom of Rheged, Knight of Freedom Award, Croatia, timeline of Kobe, Duchenne muscular dystrophy, Michigan Data Store, Ilse

Table A.15. Top 10 words for each topic trained over NBA for $k = 15$.

Topic 0	Olympic sport, Paycom Center, ESPN, Fiserv Forum, rap, NBA 2K, Kawhi Leonard, association football, Glycine max, 3928 Randa
Topic 1	basketball player, 2007 FIBA Americas Championship, Orlando Magic, Denver Nuggets, J. Walter Kennedy Citizenship Award, Sacramento Kings, Milwaukee Bucks, Naismith Prep Player of the Year Award, Indiana Pacers, Original Celtics
Topic 2	National Basketball Association, TD Garden, NBA Finals, Teledyne Technologies (Australia), Twitch, Decimus Junius Brutus, Centre d'Etudes et de Recherches Appliquées à la Gestion, Goascorán River, Sultanate of Hobyo, phenylephrine
Topic 3	entertainment, Brad Stevens, Minnesota Timberwolves, Portland Trail Blazers, Jeff Hornacek, 2019 FIBA Basketball World Cup, Major League Baseball, National Hockey League, Kobe Bryant, 2020–21 NBA season
Topic 4	team sport, Nike, Nick Nurse, Mamadama Bangoura, Lumière Award, Lyon Festival of cinema, Current concepts in the role of the host response in Neisseria meningitidis septic shock., Cyclist's palsy, Jarran Reed, Agnes Arber, monasticism
Topic 5	basketball league, NBA TV, Golden State Warriors, All-NBA Team, NBA All-Star Game Kobe Bryant Most Valuable Player Award, shooting guard, Brooklyn Nets, center, Northwest Division, 2010 FIBA World Championship
Topic 6	Basketball, Eastern Time Zone, NBA Draft Lottery, pacifism, Jackson County Ohio Democratic Party, Moulay Hassan, Wallenstein, Lan Shaomin, information leak, Plzeň-South District
Topic 7	NCAA Division I men's basketball, Los Angeles Lakers, Boston Celtics, point guard, NBA All-Rookie Team, NBA Most Valuable Player Award, NBA Rookie of the Year Award, Pacific Division, LeBron James Jr., NBA Most Improved Player Award
Topic 8	sport, Bill Russell NBA Finals Most Valuable Player Award, Ohio Mr. Basketball, Madison Square Garden, National Basketball Association Draft, gambling, Doom, Mark Daigneault, American football player, Microsoft
Topic 9	men's basketball, Commissioner of the NBA, Best NBA Player ESPY Award, Oklahoma City Thunder, Houston Rockets, 2018–19 NBA season, Dallas Mavericks, basketball at the 2020 Summer Olympics – men's tournament, Seattle SuperSonics, basketball at the 2016 Summer Olympics – men's tournament
Topic 10	ball game, Eastern Conference, Central Division, NBA Playoffs, Cryptocurrencies and Zero Mode Wave guides: An unclouded path to a more contiguous Cannabis sativa L. genome assembly, DraftKings, Ulrich Abel, Earl Grinols, Cycling Academy 2016, Wellness and health omics linked to the environment: the WHOLE approach to personalized medicine.
Topic 11	Basketball Association of America, Three-Point Contest, Tyronn Lue, Toronto Raptors, LeBron James, Philadelphia 76ers, Charlotte Hornets, NHL Hockey, New Orleans Pelicans, baseball
Topic 12	entertainment, 2014 FIBA Basketball World Cup, Southwest Division, 2006 FIBA World Championship, point forward, Doc Rivers, Shaq–Kobe feud, Tom Thibodeau, Detroit, Ohrada
Topic 13	basketball, Los Angeles Clippers, Toyota Center, Las Vegas, Category:Films shot in Serbia, Sailor Moon, Corrigendum: Nuclear RNA-seq of single neurons reveals molecular signatures of activation, Spec, Basshunter, Albert Arnold Gore
Topic 14	basketball team, NBA G League, Cleveland Cavaliers, NBA Summer League, Western Conference, small forward, power forward, Atlantic Division, NBA All-Defensive Team, Miami Heat

Table A.16. Top 10 words for each topic trained over BLM for $k = 20$.

Topic 0	2020 United States presidential election, United Nations, Donald Trump Jr., basketball, General Travelling Notes, Dining, Sunrisers Hyderabad, AVN Awards ceremony, BET Award for Best Group, José Eduardo dos Santos
Topic 1	personal finance, physical fitness, photography, 2020 United States presidential election, science, 3D film, teen pop, Republic of Ireland, Rupert C. Thompson, Mr. Burns
Topic 2	Democratic Party, Turkey, Hungary, Australia, Greece, Serbia, Afghanistan, Taiwan, Juliette Binoche, Parliament House
Topic 3	COVID-19, nonprofit organization, TikTok, Belgium, CBS, drink, Iona Preparatory School, dishonor, regulation of intracellular signal transduction, Airbus Group
Topic 4	ICC Men's T20 World Cup, cricket, online, Martin Luther King Jr., Washington, D.C., RCA Victor catalog, team sport, history, Christos Doulkeridis, iOS 9
Topic 5	United States of America, United States Congress, Texas A&M–Kingsville Javelinas men's basketball, LGBT, Bernie Sanders, Nancy Pelosi, Kevin Whately, Black Twitter, Terrence Floyd, Mehmet Âkif Hamzaçebi
Topic 6	food, George Floyd protests, cricket, hip hop music, Br'er Bear, Lord William Campbell, phenotype, freedom of the press, James Holborne of Menstrie, Orbeni
Topic 7	George Floyd, Basketball, investment, ViacomCBS, Highveld Lions cricket team, Facebook, Rhinovirus increases human beta-defensin-2 and -3 mRNA expression in cultured bronchial epithelial cells., mid-size car, Jim Stillwagon, Clinical significance of nephrotoxicity in patients treated with amphotericin B for suspected or proven aspergillosis
Topic 8	Christmas, President of the United States, police brutality in the United States, financial services, Republican Party, Mexico, Poland, Cryptocurrencies and Zero Mode Wave guides: An unclouded path to a more contiguous Cannabis sativa L. genome assembly, Ahafo Ano North Municipal District, Beaverhead County
Topic 9	anti-fascism, Canada, Egypt, retail, Bangladesh, Bulgaria, Nepal, online, Single UNIX Specification, Georgia
Topic 10	entertainment, Italy, BBC, SARSr-CoV, YouTube, Ericales, Sperm antigen with calponin homology and coiled-coil domains 1, Actaea dahurica, Phil Collins music sales certifications, Greg Prestopino
Topic 11	National Football League, rap, technology, Breonna Taylor, African Americans, athletics at the 1980 Summer Olympics – men's decathlon, Karen Berger, Association of hyperestrogenemia and coronary heart disease in men in the Framingham cohort, Jewish Encyclopedia of Brockhaus and Efron, Georg Sterzinsky
Topic 12	Black Lives Matter, NBC News, Harmelen, Élie Gesbert, James L. Farmer, Sr., Kulture Kiari Cephus, state highways in Connecticut, Vanderbilt Commodores football, ut-Ma'in, Calvini
Topic 13	George Floyd, visual arts, Twitter, Democratic National Committee, YouTube, Taiwan, Kennesaw State University L.V. Johnson Library, Luca Ceriscioli, list of people with the given name Urszula, Autumnal Cannibalism
Topic 14	Business & Finance, American football, military exercise, Zhengfang Town (Shunchang County), Services, Marris, Franz Obermeier, sports entertainment, Călărași District, team sport
Topic 15	United Kingdom, India, Russia, Ayọ Tometi, Bill Spivey, II/119 road, Antonio Marco Troiano, Jutaí Magalhães, In vitro binding of the asialoglycoprotein receptor to the beta adaptin of plasma membrane coated vesicles, Luth Enchantee
Topic 16	Joe Biden, Germany, New York City, Alma Lagoni, economy of Osaka Prefecture, Ehle, Presidential Medal of Freedom, Amharic, City of Greater Bendigo, life insurance
Topic 17	Services, Alicia Garza, pop music, Wellness and health omics linked to the environment: the WHOLE approach to personalized medicine., Patrisse Khan-Cullors, Republican Party, geography of Jan Mayen, French National Assembly, František Jiránek, guitarist
Topic 18	Black Lives Matter, NBC News, comics, Élie Gesbert, Harmelen, state highways in Connecticut, Simon, lifting equipment, James L. Farmer, Sr., Saint Pierre and Miquelon
Topic 19	Donald Trump, MSNBC, Malaysia, COVID-19, Online services., NFL Football, 1973 24 Hours of Le Mans, Nfkb1, Transmembrane protein 45B, Piratapuyo

Table A.17. Top 10 words for each topic trained over January 6 for $k = 20$.

Topic 0	Nancy Pelosi, Mike Pence, Twitter, United States senator, Paul Gosar, Jamie Raskin, soil science, .eh, Josh Hawley, Northern Pumi
Topic 1	Donald Trump, United States Department of Justice, Presidential Medal of Freedom, Ashley Biden, Federal Government of the United States of America, rap, Meliše, Rhauderfehn, Russia, Louie Gohmert
Topic 2	United States Congress, The New York Times, MSNBC, The Washington Post, United Kingdom, Nso, La Cadière-d'Azur, World Wide Fund for Nature (United Kingdom), San Fernando Mission Cemetery, 2014 Giro d'Italia, Stage 15
Topic 3	President of the United States, Elizabeth Cheney, Tucker Carlson, Hunter Biden, Matt Gaetz, Athos, Bulgaria, Franco Harris, DCR, Enniscorthy
Topic 4	Services, Vanessa Trump, Primetime Emmy Award for Outstanding Reality-Competition Program, Donald Trump 2000 presidential campaign, bibliography of Donald Trump, Ellis Island Medal of Honor, Jamaica Hospital, Trump National Golf Club, Trump National Golf Club, Kamala Harris
Topic 5	George Floyd, Federal Bureau of Investigation, Kevin McCarthy, Pietrari, Battle of Lahti, Albert Einstein, penny, multi-tasking operating system, Costa, hendiatris
Topic 6	Donald Trump, anti-fascism, 2018 United States House of Representatives elections, Bureau of Educational and Cultural Affairs, Pohjois-Espoo, Pallava script, Hip-hop theater, .wf, Vladimir Uyezd, impeachment
Topic 7	United States House of Representatives, Ivanka Trump, Donald Trump 2020 presidential campaign, Mar-a-Lago, star on Hollywood Walk of Fame, United States withdrawal from Iran Deal, Golden Raspberry Award for Worst Supporting Actor, Sean Hannity, United States of America, Dagmar Mühlenfeld
Topic 8	United States of America, White House, Desafuero of Manuel López Obrador, Cyriacus, Șicula, Deer Isle, Cynthia McFadden, Moose Jaw, 1855 census of Norway, Ter Sami
Topic 9	COVID-19, John Whitney Walter, Time Person of the Year, Ron Johnson, United States Capitol Police, Laetare Medal, personal finance, contiguous United States, Swedish Sign Language, Philip Barton Key II
Topic 10	Donald Trump 2016 presidential campaign, Fred Trump, Ted Cruz, John G. Trump, The Wharton School, Trump: The Art of the Deal, New York Military Academy, Mary Anne MacLeod Trump, Robert Trump, The Kew-Forest School
Topic 11	entertainment, Capitol Records, Steve Bannon, Lindsey Graham, Mark Randall Meadows, International Herald Tribune, Jim Banks, Wikinews article, ultrafiltration, Jacobin
Topic 12	Maryanne Trump Barry, Lara Trump, inauguration of Donald Trump, 2020 Republican Party presidential primaries, Fred Trump Jr., United States withdrawal from the Paris Agreement, Fordham University, Business & Finance, Joe Biden, Facebook
Topic 13	United States Senate, second impeachment of Donald Trump, Vice President of the United States, Chuck Schumer, retail, WikiProject European Union, Supreme Court of the United States, SkyMiles, Terracotta plate, Forest Gate
Topic 14	Joe Biden, Jared Kushner, Lauren Boebert, nonprofit organization, Johanna River, Capitoline Hill, Mueang Lamphun, digital humanities project, Flag of Ryazanovskoye Settlement, City of Penrith
Topic 15	Barron Trump, Melania Trump, Tiffany Trump, first impeachment inquiry against Donald Trump, Nagoya University, Mississippi Governor's Mansion, murder of George Floyd, political positions of Donald Trump, Bradford, 1913 Irish Open Badminton Championships – men's singles
Topic 16	2020 United States presidential election, Republican Party, inauguration of Donald Trump, Bengaluru, Frederick, Turkology, University of Namur, criminal negligence, Andres, Sarama
Topic 17	Donald Trump Jr., Washington, D.C., Democratic Party, Elizabeth Christ, Doublespeak Award, Golden Raspberry Award for Worst New Star, Jim Jordan, Jill Biden, Donald Trump, Ivana Trump
Topic 18	Eric Trump, United States Capitol, Mary L. Trump, Trump–Ukraine scandal, New York City, Trump World Tower, Ig Nobel Prize, University of Pennsylvania, Elizabeth Trump Grau, United States–Mexico–Canada Agreement
Topic 19	Donald Trump, impeachment, The Trump Organization, (E)-chlorprothixene, Bedřich Peška, canton of Reims-9, Ferrières-en-Brie, Mednogorsk Urban Okrug, Nassau County, Shannon

Table A.18. Top 10 words for each topic trained over Covid19 for $k = 20$.

Topic 0	United States of America, Turkey, Tozinameran, New Year's Day, National Football League, Doug Ford, vaccination, Ontario, UNESCO office Egypt, Tellar Prime
Topic 1	data science, Brazil, Nesmeyanov Institute of Organoelement Compounds, VMAQ-4, Klk1b22, São Pedro, Hacienda Heights, Acer truncatum, 1948–49 FA Cup, Peninsula Corridor Joint Powers Board
Topic 2	COVID-19, Giovanni Colombo, Mădulari, levomepromazine, Patrinia, surface-launched missile, Tallinn flag, computer magazine, Michel Esteban, The Tragedy of King Richard the Second
Topic 3	English, Christmas, State Council of the People's Republic of China, G. D. Searle & Company, SARS-CoV-2, Beckers, Ægir, Battle of Gabon, Taiwan, Endothelin and nitric oxide interact to regulate stretch-induced ANP secretion.
Topic 4	India, Pfizer UK, Maharashtra, severe acute respiratory syndrome, Medivation, Wuhan, Warner–Lambert, coat of arms of Lower Silesian Voivodeship, American Airlines, Punta del Este
Topic 5	technology, New Years Eve, Georges-Charles de Heeckeren d'Anthès, Jessica Lange, Mayor of Tehran, Ubiquitin-conjugating enzyme E2L 3, isoform CRA_a, boxing, High Court judge, The Jewish Home, Miyakonojō Prefecture
Topic 6	Covid Inc., Australia, Germany, Donald Trump, Pakistan, Hungary, Bangladesh, glutamatergic synapse, Comparative Genomic Analysis of Chlamydia trachomatis Oculotropic and Genitotropic Strains, Effect of endothelin-1 on glomerular hydraulic pressure and renin release in dogs
Topic 7	New Year's Eve, Israel, Russia, Philippines, science, England, Judeo-Christian Bishop of Jerusalem, Starozhily, Julian Casablancas discography, Yingtian Fu
Topic 8	SARS-CoV-2, SARSr-CoV, Pfizer, English, Taiwan, Horizon College and Seminary, Battle of Gabon, Eiffage, Girabolhos, George Bentley
Topic 9	COVID-19 pandemic, Narendra Modi, General Travelling Notes, Modelling the persistence of measles., Kosovo, government, member of Alberta Legislative Assembly, 2022 Australian Open, evidence-based policy, 1903 Tour de France, stage 1
Topic 10	COVID-19, Wellness and health omics linked to the environment: the WHOLE approach to personalized medicine., India, convention center, People's Republic of China, Vidyavati Manchu, Charlie Munger, Santa Margarita Catholic High School, Nord Stream AG, Helvetic Republic
Topic 11	Business & Finance, United Kingdom, Canada, Taiwan, South Korea, OF-1, Gender differences in the decline of mortality rates of acute myocardial infarction in West Germany., Ciaotou District, epithelial cell apoptotic process involved in palatal shelf morphogenesis, Henry Cavill
Topic 12	COVID-19, Alberta, Michel Esteban, Mădulari, Joh. Enschedé, Mowlem, Giovanni Colombo, Wikipedia:Project Swedish Academy, surface-launched missile, Utah State University
Topic 13	COVID-19, Alberta, Giovanni Colombo, Mădulari, computer magazine, Michel Esteban, levomepromazine, Tallinn flag, Wikipedia:Project Swedish Academy, Joh. Enschedé
Topic 14	entertainment, SARS-CoV-1, Spain, technology, Wyeth, whistle register, Hanazono Rugby Stadium, Chile, 1954 Tour de France, stage 15, The Football Association
Topic 15	WikiProject COVID-19, pharmaceutical industry, Services, South Africa, Cryptocurrencies and Zero Mode Wave guides: An unclouded path to a more contiguous Cannabis sativa L. genome assembly, Italy, Noubar Afeyan, Distinguishing human ethnic groups by means of sequences from Helicobacter pylori: lessons from Ladakh., Lucrezia Borgia, Sansi
Topic 16	travel, NFL Football, Ontario, African reference alphabet, Rublev, Hitzacker, Châtillon-sur-Cluses, Dumbo, Commodore 64C, PCCW Global
Topic 17	COVID-19, Alberta, Giovanni Colombo, Mădulari, computer magazine, Tallinn flag, Michel Esteban, levomepromazine, Wikipedia:Project Swedish Academy, Joh. Enschedé
Topic 18	COVID-19, Alberta, Mădulari, Michel Esteban, Giovanni Colombo, Joh. Enschedé, Wikipedia:Project Swedish Academy, computer magazine, Mowlem, Tallinn flag
Topic 19	People's Republic of China, Joe Biden, Common medications and drugs: how they affect male fertility., community health, personal finance, European Union, Poland, nonprofit organization, COVID-19, Bombay State

Table A.19. Top 10 words for each topic trained over Populist Leaders for $k = 20$.

Topic 0	New York City, Canada, Syria, Maldives, Tanzania, Switzerland, Yemen, Ecuador, Croatia, bibliography of Ghana
Topic 1	Republican Party, Elizabeth Christ, entertainment, Nancy Pelosi, California, Czech Republic, African Americans, Tri Sestry, Promise Me, Dad, London Transport Executive
Topic 2	Mar-a-Lago, Hungary, Denmark, Israel, Greece, European Union, South Korea, Australia, Iraq, Brazil
Topic 3	Melania Trump, Prime minister of India, Donald Trump 2000 presidential campaign, Bangladesh, Norway, Northern Ireland, BBC Film, Tyler County, European Union, Umbro
Topic 4	President of the United States, Democratic Party, travel, Jeremy Corbyn, The hunt of the witches, Yucatán, Xia, Melania Trump, Pennsylvania's 9th congressional district, BBC Film
Topic 5	People's Republic of China, Singapore, Holy See, Sri Lanka, United Arab Emirates, United States Senate, Sweden, Côte d'Ivoire, Finland, Mauritius
Topic 6	Washington, D.C., Donald Trump 2016 presidential campaign, White House, Hunter Biden, United States House of Representatives, telecommuting, Sekta, Tamil Nadu, Indiana Fever, Djimini
Topic 7	Narendra Modi, COVID-19, 2020 Republican Party presidential primaries, George Floyd, India national cricket team, United Kingdom, Bondum Dogon, Robert Noyce, rugby league player, Emory University
Topic 8	Narendra Modi, Hindi, Delhi, Lakshmi, United States Army infantry divisions by unit number, Raphael Trotman, MPEG-4 Part 14, NHS Test and Trace, prime minister, badminton at the 1970 British Commonwealth Games – men's doubles
Topic 9	Mary Anne MacLeod Trump, COVID-19 pandemic, personal finance, Services, Schiff, Lilla Namo, Holy See, President of the United States, Mika videography, Cyprus
Topic 10	United Kingdom, Brexit, Wellness and health omics linked to the environment: the WHOLE approach to personalized medicine., Bolivia, Jamaica, London, The New York Times, Narendra Modi, Haryana, Sistine Chapel
Topic 11	Lara Trump, Italy, Emperor Gaozu of Han, Kosovo, England, Uzbekistan, Thomas Perez, Prime Minister of the United Kingdom, Jewish–Babylonian war, United States–Mexico–Canada Agreement
Topic 12	United States of America, Brunei, United States senator, Texas, science, whistleblower, Careers, Tewksbury Township, Váha, Gabriel Lund
Topic 13	Russia, Business & Finance, National Health Service, World Health Organization, Kuwait, Seychelles, Adam Schiff, mainstream media, physical fitness, entertainment
Topic 14	2020 United States presidential election, 2016 United States presidential election, Serbia, Japan, Pakistan, United Nations, France, Turkey, Mexico, Saudi Arabia
Topic 15	India, community health, Sean Hannity, Scotland, Borys, Tripura, Member of the 30th Parliament of the United Kingdom, Outer Hebrides, Ngasa, Barranquitas
Topic 16	Boris Johnson, 2016 United Kingdom European Union membership referendum, investment, technology, sport, impeachment, Surrey Heath, Jeremy Corbyn, Maharaja Sayajirao University of Baroda, Austria
Topic 17	Donald Trump, Kamala Harris, Portland, antimycobacterial, point bar, Duchenne muscular dystrophy, Somali shilling, Jura, flag of Boston, Ecuador
Topic 18	Germany, Ukraine, Joe Biden, North Korea, Republic of Ireland, Egypt, Namibia, Tajikistan, Netherlands, State of Palestine
Topic 19	Donald Trump, Barron Trump, United States–Mexico–Canada Agreement, Gaming Hall of Fame, United States Congress, Iran, social distance, list of Award of Garden Merit dahlias, Simpson County, democratic republic

Table A.20. Top 10 words for each topic trained over NBA for $k = 20$.

Topic 0	Olympic sport, Azumino, Functional imaging with cellular resolution reveals precise micro-architecture in visual cortex., lists of office-holders, Saint-Romain-d'Urfé, Blue Hill, Avlona, Mamie Hui, Vaufrèges, Category:Maps of Ogemaw County, Michigan
Topic 1	team sport, power forward, Lance Stephenson, NHL Hockey, Nick Nurse, Online services., Rick Carlisle, 2018 FIFA World Cup, NBA Sixth Man of the Year Award, Doug Collins
Topic 2	men's basketball, Oklahoma City Thunder, Damian Lillard, water polo at the 2019 Pan American Games, Puy-Saint-Eusèbe, 2019–20 Cypriot Second Division (B Premier), physical exercise, Category:Maps of Manistee County, Michigan, Category:Maps of Fremont County, Idaho, Orange UK
Topic 3	basketball, Nairn, Uruguay at the 2018 FIFA World Cup, Felixstowe, palindromic number, Extended Display Identification Data, Solomon Sea, outline of MySQL, posterior humeral circumflex artery, men's freestyle bantamweight
Topic 4	team sport, Madison Square Garden, Yining County, Korapuzha, Padre Noguera, Development and characterization of amphotericin B nanosuspensions for oral administration through a simple top-down method, Departamento Molinos, Periodic slow earthquakes from the Cascadia subduction zone, Stewart County Tennessee Democrats, Little Duck Key
Topic 5	Los Angeles Lakers, NBA All-Star Game Kobe Bryant Most Valuable Player Award, Commissioner of the NBA, 2014 FIBA Basketball World Cup, New Orleans Pelicans, Air Jordan product line, Syracuse Nationals, Jerry West, James Borrego, 2K Sports
Topic 6	National Basketball Association, Teledyne Technologies (Australia), Rites of Spring, Joseph L. Cole, Future acceptance of adolescent human papillomavirus vaccination: a survey of parental attitudes., Brains, Wera, Lesmahagow, Goascorán River, Steropes
Topic 7	basketball player, shooting guard, Denver Nuggets, Orlando Magic, Milwaukee Bucks, Sacramento Kings, Southwest Division, Indiana Pacers, Detroit Pistons, NBA Defensive Player of the Year Award
Topic 8	sport, Bill Russell NBA Finals Most Valuable Player Award, Northwestern Wildcats women's soccer, Category:Knights Commander of the Order of Merit of the Federal Republic of Germany, Adelaide Lead, Category:Views of Elefsina, Category:Deaths in Burlington, New Jersey, Stolnici, Roberto Medina, Mayor of Blainville-sur-Orne
Topic 9	Basketball Association of America, Three-Point Contest, Best NBA Player ESPY Award, Houston Rockets, Phoenix Suns, Utah Jazz, San Francisco Warriors, National Football League, John R. Wooden Award, Joe Bryant
Topic 10	ball game, Zack Collins, Autopista AP-1, Pieter van Schooten, Rivoli, Nathaniel Seaver, rupee, Pyershamayski District, Carlos Julio Arosemena Tola Canton, Richard Woodville, 1st Earl Rivers
Topic 11	Basketball, Tokyo Big6, Gamle Oslo, Vrané nad Vltavou, acylglycerol transport, Open Data Uzbekistan, canton of Châtillon, 26 September 2020, pacifism, Streptomyces hebeiensis
Topic 12	NCAA Division I men's basketball, NBA Rookie of the Year Award, Chicago Bulls, LeBron James, Los Angeles Clippers, 2010 FIBA World Championship, baseball, Duke Blue Devils men's basketball, ESPN, basketball at the 2016 Summer Olympics – men's tournament
Topic 13	basketball, Toyota Center, Stephen Silas, Naismith Prep Player of the Year Award, gymnastics, Gary Bettman, Hautmont, Ichilo Province, Head of the House of Saud, 2001 Israeli prime ministerial election
Topic 14	Western Conference, NBA TV, Eastern Conference, NBA Summer League, All-NBA Team, Miami Heat, Pacific Division, center, 2007 FIBA Americas Championship, Northwest Division
Topic 15	Boston Celtics, NBA All-Rookie Team, NBA Most Valuable Player Award, Tyronn Lue, Crypto.com Arena, United States men's national basketball team, Kobe Bryant, National Hockey League, 2019 FIBA Basketball World Cup, Larry O'Brien Championship Trophy
Topic 16	point guard, Golden State Warriors, basketball league, Philadelphia 76ers, LeBron James Jr., basketball at the 2020 Summer Olympics – men's tournament, Ohio Mr. Basketball, retail, swingman, Michael Jordan
Topic 17	basketball team, Toronto Raptors, Washington Wizards, J. Walter Kennedy Citizenship Award, San Antonio Spurs, NCAA Division I, Major League Baseball, NCAA Division II men's basketball, NCAA Division I Men's Basketball Tournament, Naismith College Player of the Year
Topic 18	entertainment, LogicBuy, Sahitya Akademi Award in Hindi, Japanese Wikinews, Ruy Barbosa, UFC 237, 7.65×53mm Argentine, geographer, Pleasant Hill, Jodie Whittaker
Topic 19	Cleveland Cavaliers, NBA G League, small forward, Atlantic Division, NBA All-Defensive Team, New York Knicks, Nike, 2018–19 NBA season, Memphis Grizzlies, Death Lineup