Universität Zürich[UZH]

Bachelorarbeit
zur Erlangung des akademischen Grades
Bachelor of Arts
der Philosophischen Fakultät der Universität Zürich

Zero-shot Cross-lingual Transfer of the Topic Modeling Task

Verfasser: Francesco Tinner
Matrikel-Nr: 17-709-510

Referent: Prof. Dr. Rico Sennrich

Betreuerin: Dr. Duygu Ataman

Institut für Computerlinguistik

Abgabedatum: 01.12.2021

Abstract

Contextualized pre-trained embeddings have improved many domains of natural language processing. These embeddings can also be used as input representations for topic models and have increased coherence of both traditional bag-of-words based topic models and neural topic models. Multilingual versions of these embedding models have cleared the way for cross-lingual transfer of topic models using zero-shot learning. Thus, a trained topic model can be applied to texts of another language in order to predict the weights the learned topics have in each document.
Central focus is put on a neural topic model - CTM - that replaces two key components of LDA. (1) Documents are represented using contextualized sentence embeddings instead of bag-of-words representation, and (2) LDA's inference process is replaced by a black box inference method based on an adaption of autoencoding variational Bayes.

CTM will be compared in a cross-lingual evaluation task to LOTClass, weakly-supervised approach that fine-tunes a contextualized language model for category understanding, which has reached very promising accuracies in document classification tasks on various datasets. LOTClass is adapted to make it suitable for cross-lingual topic modeling. Furthermore, we will evaluate the effect certain parameters such as training epochs, the amount of supervision input, or embedding models have on the model's training dynamics, and investigate what the reasons are behind diverging predictions of document-topic distributions.

Zusammenfassung

Kontextualisierte, vortrainierte Embedding Modelle haben viele Bereiche der natürlichen Sprachverarbeitung verbessert. Diese Embeddings können auch als Repräsentation der Textdaten für Topic Models verwendet werden und haben wesentlich zur Steigerung der Kohärenz von Topic Models beigetragen. Mehrsprachige Versionen dieser kontextualisierten, vortrainierten Embeddings haben den Weg für den sprachübergreifenden Transfer von Topic Models durch Zero-Shot-Lernen geebnet. So kann ein trainiertes Topic Model auf Texte einer anderen Sprache angewendet werden, um die Gewichtung der gelernten Themen in einem Dokument einer anderen Sprache vorherzusagen, unabhängig davon auf welcher Sprache es ursprünglich trainiert wurde.
Der Schwerpunkt dieser Arbeit liegt auf einem neuronalen Topic Model - CTM -, das zwei Schlüsselkomponenten von LDA ersetzt. (1) Dokumente werden durch kontextualisierte Satz-Embedddings anstelle einer Bag-of-Words-Darstellung repräsentiert, und (2) der Inferenzprozess von LDA wird durch eine Black-Box-Inferenzmethode ersetzt, die auf einer Adaption von auto-encoding variational Bayes basiert.

Das CTM Modell wird in einer sprachübergreifenden Evaluierungsaufgabe mittels Testdokumenten in fünf Sprachen mit LOTClass verglichen, einem anderen Ansatz, der ein vortrainiertes kontextualisiertes Sprachmodell für das Kategorienverständnis anpasst. Dieses schwach überwachte Modell, erhält vor dem Start der Berechnung die Namen der Kategorien. Es erreichte sehr vielversprechende Genauigkeiten in Dokumentenklassifizierungsaufgaben auf verschiedenen Datensätzen. Wir werden das Modell so anpassen, dass es für die sprachübergreifende Themenmodellierung geeignet ist, und die mehrsprachigen Evaluierungsergebnisse mit CTM vergleichen. Darüber hinaus werden wir die Auswirkungen bestimmter Parameter wie Trainingsepochen, die gegebene Anzahl Wörter pro Kategorie oder die Verwendung verschiedener vortrainierter Embedding Modelle auf die Trainingsdynamik des Modells evaluieren und untersuchen, was die Gründe für divergierende Vorhersagen von Dokument-Themen-Verteilungen sind.

Acknowledgment

First and foremost, I want to thank Duygu Ataman for all the interesting discussions we had, all the guidance you gave me and your continuous support, despite university changes, and the irks and quirks of time zones. Also, I want to thank Yu Meng and Federico Bianchi for answering any questions I have had. Lastly, I want to give heartfelt thanks to Duygu Ataman, Andrea Jurt Massey, Dylan Massey, and Rico Sennrich for providing feedback on earlier drafts of this thesis.

Contents

List of Figures

List of Tables

List of Acronyms

NLP	Natural Language Processing
LDA	Latent Dirichlet Allocation
ProdLDA	Product of Experts LDA
BERT	Bidirectional Encoder Representations from Transformers
sBERT	Sentence BERT
mBERT	Multilingual BERT
NSP	Next Sentence Prediction
TLM	Translation Language Modeling
RoBERTa	Robustly Optimized BERT Pre-training Approach
BoW	Bag-of-Words
BPE	Byte Pair Encoding
KL	Kullback–Leibler (Divergence)
NPMI	Normalized Pointwise Mutual Information
CTM	Contextualized Topic Model
VAE	Variational Autoencoder
AVITM	Autoencoded Variational Inference for Topic Model
LOTClass	Label Names Only Text Classification
MLM	Masked Language Modeling
MCP	Masked Category Prediction
PCA	Principal Component Analysis
t-SNE	t-distributed Stochastic Neighbor Embedding
MDS	Multidimensional Scaling
GAN	Generative Adversarial Network
NLTK	Natural Language Toolkit
POS	Parts-of-Speech

1 Introduction

1.1 Motivation

During my undergraduate studies I was confronted and fascinated by topic models and their ability to find the underlying topic gist of large document collections. The details behind the probabilistic topic models were often only touched upon briefly, as the focus was set on applications, such as topic models as part of recommendation systems, or as tools to automatically visualize, organize, and summarize large collections of documents.
Last semester I had the opportunity in a module to do a final project on the task of topic modeling where I had to restrict the thematic focus to LDA[1] topic models. Shortly after starting that project, it was clear that I needed to expand it into a thesis in order to focus on more recent approaches and applications.

1.2 Main Objective

This thesis will evaluate two exemplary approaches to topic modeling that rely on contextualized embeddings to represent the collection of documents and the tokens they contain. These embeddings have improved coherency of both traditional bag-of-words topic models and neural topic models[Bianchi et al., 2021a]. We will use multilingual contextualized pre-trained embeddings as document representation. As they are not language specific, but rather language independent or transcendent, the numerical encoding of a text in one language will be very similar to the encoding of a parallel text in another language. This attribute of multilingual contextualized embeddings allows the application of a trained topic model to another language. We will evaluate two topic models in a cross-lingual setting, and investigate their training dynamics, and the effects different embedding models have during evaluation. We will be following an evaluation approach based on Bianchi et al. [2021b].
Our focus lies on two approaches: (1) CTM a neural topic model which applies

[1]Latent Dirichlet Allocation (LDA)

Variational Autoencoders to the topic modeling task and (2), LOTClass, a weakly-supervised approach to document classification, that fine-tunes a BERT model for category understanding. We are not aware of approach (2) being ever evaluated in a cross-lingual setting.

1.3 Structure

This thesis can be broadly divided into two parts. The first part contains background explanations regarding the key principles CTM and LOTClass rely on. In a second part we explain Bianchi et al.'s approach to automatic evaluation of topic models in a cross-lingual setting, conduct our own evaluations and derive a comparison between the two different methods.
At the beginning of this thesis we will present the techniques used to convert language into a numerical representation, incorporating both the context and the semantics of language. These mappings are not static, but words can be encoded differently based on their context which is used to derive their meaning. This proved to be valuable in the resolution of some ambiguities. We will especially focus on multilingual models, since they allow us to apply zero-shot learning. In this setting, the topic model - after being trained on a monolingual dataset - is applied on a language never seen before.
In Chapter 3 we will quickly explain generative probabilistic topic models and their most prominent model LDA. Then we will present neural approaches to topic modeling. An explanation of the datasets, where we are going to apply the topic models on follows. We will focus on the different attributes of the various datasets and their suitability for the task of topic modeling and document classification. Then, we will evaluate how successful the transfer to another language was by applying the model on parallel test sets in five languages. Evaluation is always done in a manner of comparison between the English test set and the test set in another language, e.g., Portuguese. We will evaluate to what degree the top-predictions of each document pair match. Furthermore, we will also account for partial matches, by calculating the average divergence between the document-topic distributions of two parallel test sets. These multilingual evaluation metrics are calculated for both models. We will evaluate the effect of varying model parameters such as the number of training epochs, the amount of supervision input, or the choice of embedding model have on the model's training dynamics and on the results of the multilingual evaluation metrics. Further, we will investigate what the reasons are behind diverging predictions of document-topic distributions.

2 Text Representation Techniques

2.1 Language Representation

The content of textual documents needs to be mapped to numerical representations in order to be processed further. The most basic text modeling technique, Bag-of-Words (BoW), and a state-of-the-art method, pre-trained contextualized word embeddings will be presented.

2.1.1 BoW Document Representation

Text features can be represented by counting the occurrence of known words within a document. These known words belong to the vocabulary that needs to be pre-defined. Information about word order or place of occurrence is not being considered, which can be a disadvantage for certain applications. Using the BoW feature extraction method, we can represent a document through a vector that has the same dimensions as there are tokens in the vocabulary. Each dimension corresponds to a word in the vocabulary and the value holds the corresponding word count. This approach is used in most LDA topic models as document representation technique [Blei et al., 2003].

2.2 Pre-trained Neural Language Representations

In the following sections, we will mainly focus on discussing BERT and the training processes involved to derive the pre-trained models that we will use later to generate document embeddings. Additionally, we will provide a brief overview of related BERT based models that improve aspects of the BERT-base version and allow to derive embeddings more suited for certain applications. These models include Sentence-BERT, multilingual-BERT, RoBERTa and DistilBERT.

2.2.1 BERT

BERT is an abbreviation for Bidirectional Encoder Representations from Transformers. The Transformer architecture is used in various NLP applications, such as machine translation or language modeling. In contrast to language models based on other architectures, where text sequences could only be read sequentially from left to right or vice-versa, bidirectional training using self-attention allows reading the entire sequence at once in a non-directional way. In essence this architecture allows a deeper sense of the language context and results in more fluent language representations. Also it is parallelizable which leads to reduced training time [Vaswani et al., 2017; Devlin et al., 2019; Raaijmakers, 2019].
The attention mechanism allows the learning of contextual relations between tokens, enabling the model to focus specifically on other, possibly connected words while encoding a certain token. Attention allows these clues to be considered and can help to lead to a better encoding. E.g., in a sentence containing a coreference, attention allows us to consider the entity, while encoding the pronoun, that way, the encoding of the pronoun can contain references to the referred object.

BERT models can be utilized to represent the meaning of tokens through contextualized embeddings. A BERT model can be used to encode tokens into a dense, contextualized numerical representation. These embeddings can then be employed on various downstream tasks. Tokenization of input documents is performed before learning a character-based Byte-Pair-Encoding (BPE) to guarantee a heuristic tokenization for all input documents. Byte-pair encoded vocabulary consists of a mixture between characters and word level representations which are obtained by statistically analyzing the corpus. This allows for segmentation of rare words into smaller meaning-bearing units. A vocabulary size of 30'000 subword units is chosen and the vocabulary is learned after the input has been preprocessed. A numerical ID is assigned to each of these tokens in order to transform text into a numerical input format suited for neural networks. Additionally, to incorporate word order, a positional encoding is added to a token's embedding to consider the order of the sequence. Otherwise positional information of the input tokens could not be considered [Sennrich et al., 2016; Wu et al., 2016; Devlin et al., 2019].

A BERT-base model such as *bert-base-uncased* generally consist of 12 encoding layers, that each have a hidden size of 768 neurons. 12 attention heads per layer allow to consider 12 distinct aspects of other tokens during encoding. This model thus contains 110 million trainable parameters. The same architecture can be used for

various NLP tasks, some require adding another layer e.g., for classification. In comparison a BERT-large model consists of 24 hidden layers with hidden size of 1024, which results in three times more trainable parameters than the base version [Devlin et al., 2019].

2.2.1.1 Training Process

When training a language model, it is difficult to define the prediction goal. Often the task of next word prediction is used, which is a directional approach, and thus not optimal for BERT, instead two non-sequential training strategies are used: masked language modeling (MLM) and next sentence prediction (NSP), which we will explain in the next few sections. The model is trained on both tasks simultaneously; the loss of both tasks is combined and must be minimized during training on a very large corpus.

MLM Before applying the model to a text, a percentage of tokens is erased from the original text and replaced by a special [MASK] token. The task of the model during MLM is then, to predict the word that was replaced by the mask token, by using the context of both sides as guidance. This prediction can be derived by passing the encoder output through a classification layer. The output vectors must then be transformed to match the vocabulary dimension. This can be achieved by multiplying the output of the classification layer with the embedding matrix. In order to derive the probabilities of occurrence for each word in the vocabulary (at the position of the mask token), the softmax activation function is applied on each output node's value [Raaijmakers, 2019].
In practice, not all tokens that are masked are indeed replaced by [MASK]. 15% of all tokens occurring in the training corpus are masked out, of those masked words, 80% are replaced with the [MASK] token, the remaining 20% are either replaced with a random different token, but not the one that was masked out, or left unchanged. This split was determined on a trial-and-error basis and is done to ensure the model learns more about the entire input and does not only consider the current input token [Devlin et al., 2019].

NSP Sometimes next sentence prediction (NSP) is also referred to as sentence similarity prediction. This constitutes the second objective during training of a BERT model. The task here is to predict whether two sentences are a subsequent pair. Therefore, the model needs to be able to distinguish the two sentences, i.e., sentence start, and end need to be marked. For that reason, [CLS] and [SEP] tokens are

inserted into the sentence pair such that the input is of the following form:

[CLS] Sentence 1 [SEP] Sentence 2 [SEP].

[MASK] tokens could be present in the sentences, since the two training strategies are being carried out simultaneously. For the purpose of simplicity, they were omitted here.

For each token, additional embeddings are added that depend on a token's position and sentence affiliation (only two possibilities: first sentence or second sentence). The sequence, including special tokens, is passed through the transformer model, the output corresponding to the [CLS] input token is transformed using a classification layer with softmax activation function to obtain the prediction of the two sentences being neighbors [Raaijmakers, 2019]. Before using the encoded [CLS] token representation, the model should be fine-tuned on domain specific data or task, only then, this token becomes meaningful as a sentence vector for sequence classification. A pre-trained BERT model can be fine-tuned to many NLP tasks, such as question answering, sentiment analysis, and translation. In 2018, BERT obtained state-of-the-art results on eleven NLP tasks [Devlin et al., 2019].

2.2.1.2 Sentence-Transformers (sBERT)

A standard BERT model can be used to measure the similarity between two sentences using the encoded output at the respective position of the [CLS] input token. BERT can only compare two sentences. In tasks, where the objective is to find the most similar sentence to a given phrase, we need to pass all possible combinations through the BERT model and use a classifier or regressor to obtain the relatedness of all possible pairs of sentences. For n sentences this would result in $\frac{n(n-1)}{2}$ forward passes, which makes BERT useless for applications like clustering. Using $n = 10'000$, finding the most similar sentence would result in approximately 50 million inference computations. Also in semantic search, where we compare the input query with each available sentence (n) in the dataset, we would need n forward passes [Reimers and Gurevych, 2019].
Sentence-BERT (sBERT) - which is adapted to the task of sentence-similarity detection - solves that limitation by computing dense, fixed sized embeddings for each sequence separately. Inputs of variable length are always mapped to a dense representation of a fixed size. Any of these representations can then be compared in terms of semantic similarity to any other embedding using cosine similarity.

When encoding a sentence to a numeric representation, an additional layer is added, which condenses all embedded tokens to a sentence representation of a fixed size (unaffected by the varying input lengths of input sequences). E.g., a mean pooling layer takes each token's embedding vector and calculates the average embedding vector (elementwise) for a sequence. Reimers and Gurevych evaluated various pooling strategies: mean and max pooling or using the [CLS] token that the BERT model generates by default. Their research shows that using BERT's [CLS] token or averaging BERT's word embeddings performs worse than the mean- and max pooling of sBERT. Especially in the task of predicting semantic textual similarity between two sentences. This is due to the fact that sBERT was fine-tuned on semantic textual similarity data, in contrast to BERT which is only pre-trained on the two tasks mentioned in Section 2.2.1.

The sBERT architecture during fine-tuning is different from standard BERT as it consists of two tied BERT networks. They share all parameters, hence the term "Siamese BERT Networks". There are different ways to fine-tune sBERT, and thus different loss functions are involved. In a supervised setting - the simplest setting - where labels about the similarity of two sentences are available, sentences A and B are passed through the network, resulting in the two sentence embeddings, of which cosine similarity is computed and compared to the gold similarity score. Mean-squared-error loss (between the predictions and the gold labels) is then used as the regression objective function.

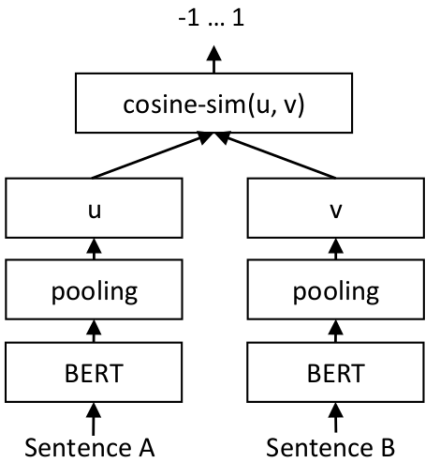

Figure 1: sBERT architecture in a setting of fine-tuning on labeled data. Source: [Reimers and Gurevych, 2019, 3]

In the unsupervised case - where no annotation labels are available - the classification objective function is used, and cross-entropy-loss is optimized [Reimers and Gurevych, 2019].

2.2.1.3 Multilingual BERT Versions

Multilingual BERT (mBERT) supports 104 input languages. This model was trained with the MLM objective on Wikipedia texts in 104 languages with a shared vocabulary across all 104 languages. The smaller version (*bert-base-multilingual-uncased*) will be used in Section 6.2.3 [Devlin et al., 2019].

XLM is like mBERT also trained on the MLM task, additionally it was trained with a translation language modeling (TLM) objective, to encourage the model to learn similar representations for different languages. Translated sentences, or a multilingual parallel dataset must be available. A dataset commonly used for this purpose is *XNLI*[1]. During MLM with the TLM objective, sentences are masked and the model is allowed to use tokens from other languages to predict the masked tokens [Conneau and Lample, 2019].

XLM-R is a self-supervised *RoBERTa* model trained on a total of 2.5TB of text in 100 languages. This unlabeled training dataset is the largest dataset used to train any of the presented models. It was extracted from *CommonCrawl* datasets[2]. The model is trained using only the MLM objective as is usual for RoBERTa models. Later, we will use this fine-tuned version - *paraphrase-multilingual-mpnet-base-vs* - to represent sentences Liu et al. [2019]; Reimers and Gurevych [2019, 2020].

2.2.2 RoBERTa

RoBERTa is an acronym standing for Robustly optimized BERT approach. It optimizes the training phase of BERT to reduce time during pre-training.
During pre-training the NSP objective is eliminated. RoBERTa is trained on larger batch sizes and can encode larger input sequences. Furthermore, larger batch sizes proved to be advantageous to the MLM objective. Another modification of the MLM training step is that the masking pattern can change dynamically. In contrast to BERT, where the masking is static and only performed once.

[1] Cross-lingual Natural Language Inference dataset [Conneau et al., 2018]

[2] `https://commoncrawl.org/`

RoBERTa was trained on the same datasets as BERT, with the addition of large collections of recent English news articles, web content texts obtained from links to webpages found in Reddit posts and a story dataset from *CommonCrawl* [Liu et al., 2019].

A byte-level BPE vocabulary of 50'000 units was chosen. This approach uses *bytes* instead of Unicode characters as subword units and does not need any preprocessing or additional tokenization of the input. While this approach does slightly worsen performance, it provides a universal encoding scheme [Liu et al., 2019].

2.2.3 DistilBERT

DistilBERT is a smaller and lighter pre-trained BERT model that uses knowledge distillation during pre-training to reduce model size in comparison to BERT by 40% while maintaining comparable language understanding capabilities. Additional advantages in comparison to the standard BERT model include faster and cheaper training. This model was the basis for *distiluse-base-multilingual-cased-v1* [Sanh et al., 2019].

Knowledge distillation is a compression technique using a smaller model (the student model) and a larger model (the teacher model). The aim is that the student model can reproduce the behavior of the larger model by mimicking the full output distribution of the teacher model. It is required to use the whole distribution and not only the top prediction (using one-hot-encoding of the target class) in order to transfer the knowledge of the teacher model to the student.
DistilBERT is based on the architecture of BERT, except for the number of encoding layers that is reduced to 6 [Sanh et al., 2019].

3 Topic Models

Topic models are a subtype of mixed membership models, from the domain of statistics. Applying a topic model to a large collection of texts allows the discovery of thematic structure, and to annotate the documents according to their most dominant topic, which relates topic modeling strongly to document classification. Each topic consists of a group of words associated under a theme. Expressed is this concept as a distribution over terms of a fixed vocabulary. The distribution of topics over a given document or the entire collection of documents can be inferred and can further be used to visualize, organize, and summarize a document collection. Extensions to the model and relaxation of assumptions allow for other applications such as the automatic annotation of pictures [Blei, 2012; Blei et al., 2003]. We will first illustrate the difference between generative models and discriminative models before explaining LDA and neural topic models.

3.1 Generative Probabilistic Models

Generative modeling in contrast to discriminative modeling aims at solving the general problem by learning a joint distribution over all the variables. This simulates how data is generated in the real world, by expressing causal relations of the world. Discriminative modeling on the other hand aims at learning a predictor given the observations. I.e. an observation is given in the form of the BoW[1] representation of a document. This vector of word counts can then be used to map a document to a category. In other words, we learn a mapping in the same direction as we intend to make predictions in the future. In generative models, this is done the opposite way even though we can use Bayes rule[2] to convert the generative model into a predictor [Kingma and Welling, 2019].
Data is treated as observations that originated from a generative probabilistic process that includes hidden variables - which are often referred to as latent variables.

[1] Bag-of-Words, for an explanation see Chapter 2.

[2] Formula is given in Section 3.2.1.1.

In text documents the hidden variables represent the thematic mixture of a document from the collection. Assumptions in mixed membership models are, that each document is coming from an individual mixture of topics, but topic components are fixed across all documents [Blei, 2012].
The use of posterior inference, allows to learn the topics that best describe the given collection of text documents by calculating the conditional distribution of the hidden variables (topics), given the observations[3]. Having calculated or approximated the posterior, allows new data to be situated into the estimated model and to obtain the topic mixture of a new document [Blei et al., 2003; Blei, 2012; Steyvers and Griths, 2006].

3.1.1 Generative Process

The generative process is purely imagined and only needs to make sense for the target at hand. Its direction, in comparison to discriminative processes, is reversed, hence we are given the categories and generate documents, based on the category's term distributions. This process does not need to generate readable, coherent documents and is only an assumption as to where the data of the model originated from.
The generative process of probabilistic topic models involves four steps:

1. A distribution over all topics is chosen. The form of this specific distribution can take various forms depending on the document.
2. For each word in the document, a topic from the distribution is chosen.
3. For each topic, a word associated with the topic is drawn.
4. Steps 1-3 are repeated for every word. This illustrates the assumed way of how each document was created.

3.1.1.1 Posterior Inference Approximation

To infer the underlying topic structure[4], we need to calculate the posterior. All these hidden distributions were mixed to form the collection of documents. It is assumed that for every data point, there exists a corresponding local latent variable. In a simple model we can infer the posterior distribution of the latent variables, conditioned on the observed probabilities by applying Bayes' rule. In more complicated

[3]Observations are in the case of BoW text feature representation, the word counts of a document.

[4]The underlying topic structure consists of all hidden variables such as the topic proportions (weights) associated with each article or a topic's distribution over terms of the vocabulary

models, Bayes' rule is intractable, and we need to approximate the intractable distribution with a simpler distribution. Various methods to approximate the posterior distribution exist: e.g., Gibbs sampling or variational inference which are algorithms that transfer the inference problem to an optimization problem. By calculating the distance (measured with Kullback-Leibler (KL) divergence) between different families of probability distributions and the hidden structure, they allow finding the distribution closest to the posterior [Blei et al., 2003; Blei, 2012].

3.1.2 Latent Dirichlet Allocation (LDA)

LDA is the most prominent generative Bayesian probabilistic model in the domain of topic modeling. It assumes that documents contain multiple topics of different proportions. A restriction is that the number of topics must be set in advance. The same set of topics is shared by all texts in the collection, only the weights of the topics vary. Through changing these topic weights, a specific document can be generated. Defined is a topic as a distribution over a fixed vocabulary, this distribution is assumed to exist even before the texts were generated. This generative, random process is purely imaginary and explains intuitions behind the model. The model assumes each document was generated: First, for each document a form of distribution over all topics is chosen. In contrast to Section 3.1.1 in LDA the Dirichlet distribution is used to represent the distribution over topics in the document. Secondly, for each word a topic assignment is chosen, and thirdly, a word is sampled from the chosen topic [Blei et al., 2003].

Observable is only the text, while the topic structure, and per-document topic distributions are hidden. The problem in the inference process when calculating a topic model is using the observed documents to infer the (hidden) topic structure that is most likely to describe a certain text. This is done through a search over the topic structure, by approximating the conditional probability distribution of the topic structure given the observed documents (called posterior) [Blei et al., 2003]. We can express the document-topic probability distribution as a multinomial distribution over the topics [Bianchi et al., 2021b].

3.1.2.1 Limitations

The number of topics in the collection is assumed to be known and cannot be changed. Bayesian nonparametric topic models can overcome that limitation as

they allow for unseen topics. Furthermore, texts are represented as BoW using only word frequencies to represent a document in terms of its word counts. Thus, word order, or context of words is not accounted for. Relaxing the constraints of LDA and allowing for permutation of the words in a document, leads to improved language modeling performance. Another limitation is that the chronological order of the documents is not considered. Dynamic topic models take into account that topics change over time and make this change visible [Blei, 2012].

3.1.2.2 Incorporation of Meta-data

Additional information can be of help when fitting a topic model to a collection of documents. Meta-data such as author information can be integrated into the generative process, which allows to consider additional properties of the text collection into the derivation of topics. An example of such an adapted topic model is the author-topic model. Use of Dirichlet-multinomial regression allow inferences about both documents and authors, which makes computation of similarity between two authors possible [Rosen-Zvi et al., 2004].

3.2 Neural Topic Models

Neural networks can be leveraged to improve performance, usability, flexibility, and efficiency of topic models. Moreover, neural networks can be used for NLP tasks - where traditional topic models are hard to apply - such as text generation, translation, and document summarization [Zhao et al., 2021].

3.2.1 Variational Autoencoder (VAE)

We will start by explaining the intuition behind the concept of VAEs.
VAEs are often employed for the purpose of unsupervised representation learning but can also be applied in a generative way to create new data, as VAEs have the ability to generate new examples similar to the dataset they were trained on. VAEs in a broader sense are a useful architecture for "[...] finding disentangled, semantically meaningful, statistically independent and causal factors of variation in data [...]" [Kingma and Welling, 2019, 4].

A VAE consists of two connected, but individually parameterized models that sup-

port each other - a recognition model (or encoder) and a generative model (or decoder). According to Bayes rule (see Section 3.2.1.1), the encoder model is the approximate inverse of the decoder network. The output of the encoder model is used to reconstruct the input data with minimal loss [Kingma and Welling, 2019]. We will explain the loss function in a more detailed way in the next Section (3.2.1.1). Each dimension in the latent space represents a probability distribution for each attribute learned from the data, while the output of the encoder network is always a single value sampled from each dimension of the latent space. The decoder network takes these values one after the other and tries to recreate the original input.

An intuitive example of a use case of VAEs is image re-synthesis. Training a VAE model on a large dataset of pictures of faces allows the autoencoder in the ideal case, to learn descriptive attributes of these faces and the ranges these values can be in. Each attribute is represented as a probability distribution in the latent space that the encoder network generated. By estimating a probability density function of the training data, very uncommon instances will be assigned a low probability value, since the example differs much from the training data. In our example of faces, the learned latent variables/attributes could be hair color, skin tone, gender, glasses, and beard, which can be used to describe the picture. The encoder samples from each latent state distribution a number, and outputs a vector that contains the sampled value for each latent distribution. Then, based on that sample vector, the decoder network can (re)generate the original image. We note that by changing the vector manually we could modify the picture that will be generated by the decoder network.

The encoder model outputs a distribution of possible values for each latent dimension, from which a random value is sampled. This is done to derive a continuous, smooth latent space representation of the input data's attributes. The decoder model is trained to reconstruct the input given any sampled value from that distribution. Thus, values that are close in the latent space will correspond to similar reconstructions [White, 2016; Kingma and Welling, 2019].

3.2.1.1 Inference Process

Inference is the process of deducting properties about a probability distribution of the given data. Thus, we want to approximate properties of the population by analyzing a data sample. The inference process of the autoencoded variational

framework resembles LDA's inference process. We assume to have a hidden variable in the latent space z. This variable generates an observation or attribute x through use of the decoder network. Note that x is observable but we want to infer the characteristics of the hidden variable z, this is expressed by $p(z|x)$ [Goodfellow et al., 2016, 693-698]. Where $p(z)$ is the prior distribution that represents beliefs, that we have about the true value of the parameters. The prior distribution beliefs are as in LDA assumed to follow a Dirichlet distribution[5]. We can apply Bayes' theorem to obtain the posterior distribution:

$$p(z|x) = \frac{p(x|z)p(z)}{p(x)},$$

that represents our beliefs about the parameter values after having taken observed data into account.

The term $p(x)$ - sometimes referred to as evidence - of the posterior distribution is defined by:

$$p(x) = \int p(x|z)p(z)\, dz$$

but we must state that the exact calculation of $p(x)$ is intractable. Thus, variational inference is applied in order to estimate the probability of our observed attribute $p(x)$. The posterior distribution $p(z|x)$ can be approximated by another traceable distribution $q(z|x)$, by learning the parameters such that this distribution is a close approximation to the posterior. We can use KL divergence to measure the difference between $p(z|x)$ and $q(z|x)$ and choose our approximated distribution $q(.)$ as close as possible to the original distribution by minimizing KL divergence between $p(z|x)$ and $q(z|x)$. In the following formula this objective is rewritten as a maximization problem using variational inference [Blei et al., 2016]:

$$E_{q(z|x)} \log p(x|z) - KL(q(z|x)||p(z)).$$

The second part of the formula will implicitly be minimized and the approximated distribution $q(z|x)$ will be chosen such that it is as similar to the intractable distribution. The first part of the formula represents the likelihood of reconstruction of

[5]Use of Dirichlet prior is important in topic models to obtain interpretable topics, but this prior cannot be used for inference using a VAE model. Thus, it is approximated by a Laplace approximation, which in essence allows to use a Gaussian distribution as prior [Srivastava and Sutton, 2017].

the original input x given the latent variable z [Kingma and Welling, 2019].

We have used $q(.)$ as an approximation to infer the possible hidden latent states that were used to generate the observation x. This can be modeled using a neural network. An encoder network learns the mapping from observation to latent state and a decoder network learns the mapping from latent state to the observation. We measure (1) The reconstruction error through maximizing reconstruction likelihood and (2) KL divergence of the learned distribution $q(z|x)$. The loss function is constructed such that it discourages reconstruction error and encourages the learned distribution to be close to the true prior distribution $p(z)$ [Rezende et al., 2014; Kingma and Welling, 2014, 2019; Blei et al., 2016].

VAEs can also be used for the topic modeling task, as shown by Srivastava and Sutton [2017]. We will explain their adaptation of LDA in Section 3.2.2.

In addition to the neural variational framework, other frameworks such as autoregressive models, generative adversarial nets (GANs), graph neural nets, and attention based neural networks can be used for inference [Kingma and Welling, 2019].

3.2.1.2 Undercomplete Autoencoders

Data compression is a usage case, where a self-supervised neural net encodes or extracts useful features from the input data and stores them in a dense latent space representation, which reduces the transmitted data. In the context of autoencoder models, a dense representation is learned for input data, while simultaneously the reconstruction from the latent space into the original input data is learned. Aim is the minimization of the reconstruction error. This constitutes a supervised learning problem, where the original input represents the labels [Kingma and Welling, 2014; Rezende et al., 2014; Srivastava and Sutton, 2017].

One simple way of achieving a dense latent representation is through undercomplete autoencoders, where the number of nodes in the hidden layer (one of the layers between input and output layer) of a neural network is scaled down. This limits the flow of information through the network, ensuring that useful properties of the input data are learned, and prevents copying of input data to the output [Rezende et al., 2014].

3.2.2 ProdLDA Model

ProdLDA is an extension of LDA, based on the neural variational framework. It replaces the inference process of LDA with an inference method for LDA topic models (AVITM) based on autoencoding variational Bayes to approximate the posterior distribution of the observed documents. This model calculates more coherent topics than LDA and training time is improved. AVITM is a black box inference method that does not need to be customized to the exact topic model used and can handle changes in the model without rigorous mathematical derivations [Srivastava and Sutton, 2017]. The inference method is based on a weighted product of experts, which allows it to directly map documents to an approximate posterior distribution. In comparison to LDA, the model can make predictions that are sharper than the components that are being mixed. A further advantage is, that the mapping is "well-behaved", a small change in the document results in a small change in the topics only [Srivastava and Sutton, 2017].

3.2.2.1 Implementation of Autoencoded Variational Inference for LDA Topic Models

The encoder network of the VAE generates outputs describing a distribution for each dimension of the latent space. Assumed is that the prior distribution $p(z)$ follows a Gaussian distribution (that can be described by μ and σ), thus the output of the encoder network describes the distribution of an attribute of the latent space using mean and variance of a normal distribution - in contrast to LDA where the Dirichlet distribution is used. To define how the dimensions are correlated a covariance matrix is used.
The decoder network on the other hand will than generate a latent vector by sampling from the received distributions of the encoder and reconstruct the original input text.
Sampling from the distribution needs to be adapted, as sampling from a parameterized distribution does not allow gradients to backpropagate. For backpropagation to work, all operations must be differentiable. A transformation that substitutes the approximated distribution q to a parameterless random variable is needed such that sampling is done from a standard Gaussian distribution. This is called the "reparameterization trick". That shifts the standard normal distribution by the latent distribution's μ and scales it with σ [Srivastava and Sutton, 2017; Blei et al., 2016;

Bianchi et al., 2021b].

ProdLDA uses BoW to represent the input texts. Thus, it is required to preprocess the corpus and to keep only a limited vocabulary of important words, e.g., only certain parts-of-speech (POS) are kept such as nouns, and proper names, as they provide most content in order to interpret topics well.

Bianchi et al. [2021b] will modify ProdLDA such that contextualized word embeddings can be used as input representation. This model is called Zero-Shot Cross Lingual Contextualized Neural Topic Model (CTM).

3.2.3 Zero-Shot Cross-Lingual Contextualized Neural Topic Model (CTM)

CTM is an extension of ProdLDA a neural topic modeling approach based on VAE [Srivastava and Sutton, 2017; Kingma and Welling, 2014]. Bianchi et al. replace in CTM the BoW input representations used in Srivastava and Sutton's ProdLDA with contextualized pre-trained (multilingual) embeddings. Pre-trained embeddings contain more information about the input texts than the traditional BoW input representation, such as external knowledge which allows for disambiguation, sentence similarity, and especially incorporation of word order. This is due to the inherent sense of pre-trained contextualized embeddings. Furthermore, the use of multilingual contextualized pre-trained embeddings enables zero-shot cross-lingual topic modeling for unseen languages. The prediction of topic weights on test documents in unknown, unseen languages becomes possible by transferring the pre-trained topic model, under the condition that the language is also supported by the pre-trained multilingual embedding model. This is not possible for BoW embedding based approaches to topic modeling, since text representations cannot be transferred easily into other languages [Bianchi et al., 2021b].

As in ProdLDA, an inference network maps the representation of an input document onto a latent representation, that is then used in a second step by a decoder network with the objective of reconstructing the BoW input representation. Also, in the case of CTM, the reconstruction is in the BoW format containing the vocabulary of the training language, and not in the format of the original embeddings from e.g., sBERT. All articles of the W2 dataset were clipped to a maximal length of 200 tokens. In order to prepare the BoW dataset a simple preprocessing method

was used that removed stopwords, and created a vocabulary of the 2'000 most occurring tokens. This step is done in addition to the preprocessing and tokenization involved in each pre-trained BERT based model [Bianchi et al., 2021b] which reduce the length of input documents sometimes even further to a maximal length of 128 tokens[6].

3.2.4 Fine-tuning BERT for Category Understanding (LOTClass)

This method constitutes this thesis' second area of focus. LOTClass is not based on an LDA topic model, but on the way a human being would approach classification of texts into categories. In a document, a human only needs to see some words that are indicative for a certain category (e.g., MBS, options, EBIT, dividends) and on that basis this newspaper article is categorized into the category financial news.
LOTClass[7], is a weakly-supervised method [Meng et al., 2020]. The objective of the model is classification of articles into categories based on a few given representative words per category (the supervision input). All input label words can only be associated with exactly one category. In contrast to semi-supervised methods, there are no ground truth category labels needed to guide the training process. Instead, the model learns to understand the label names and can predict the article's category label based on the occurrence of certain words that are judged as category indicative. As a knowledge source for category understanding, BERT, a pre-trained contextualized language model, is used[8]. Advantages of that transformer based language model are a strong feature representation, and the ability to capture long range dependencies in texts [Meng et al., 2020].

3.2.4.1 LOTClass' Training Process

Three stages of training are involved. In a first step, the model learns to understand the input categories. Semantically related words are associated with the label names given as weak-supervision input. For each category, a vocabulary is constructed that contains semantically associated words with the provided label names. These words are masked out and the masked language modeling objective (MLM) is used to predict, based on the context before and after the [MASK] token, what words can replace the category label names in most contexts. Output of the language model

[6]These differences concerning limitations of input length are explained in Table 5.

[7]Label-Name-Only Text Classification. Source code available at `https://github.com/yumeng5/LOTClass`

[8]Detailed explanations of BERT can be found in Section 2.2.1

indicates the likelihood of occurrence of every word in the vocabulary at the position of the masked word. The 50 words considered most likely to replace the original word (in most contexts) are appended to the label names of that category [Meng et al., 2020]. This fine-tunes the pre-trained BERT model to adapt to the specific attributes of the dataset.

In a second step, all occurrences of words in a category's vocabulary are found in each document and the model is trained to predict the implied categories of these words. We cannot search for all occurrences of the words in a category's vocabulary, because occurrence of such a word must not in all cases hint at a certain category. Often meaning is ambiguous, and we can resolve such ambiguities by considering the context of category indicative terms. This challenge is addressed by masked category prediction (MCP). Instead of only predicting replacement words for all occurrences of words in the category vocabulary, more words (not only category vocabulary words) are masked out and a pre-trained language model is used to find the 50 most probable replacement words. If more than 20 of the 50 replacement words are found in the vocabulary of a certain category, the word that was masked out, is considered as category indicative. Note also, that we obtain a set of category representative words, along with the label of the corresponding category. This is used to create supervision in a setting of unlabeled data.
Each category representative word w is then masked out, and the model is trained to predict the category that w is indicating. This is achieved using cross-entropy loss and an additional linear layer as classifier, which is used to assign the contextual embeddings of the masked word to the most suiting category [Meng et al., 2020].

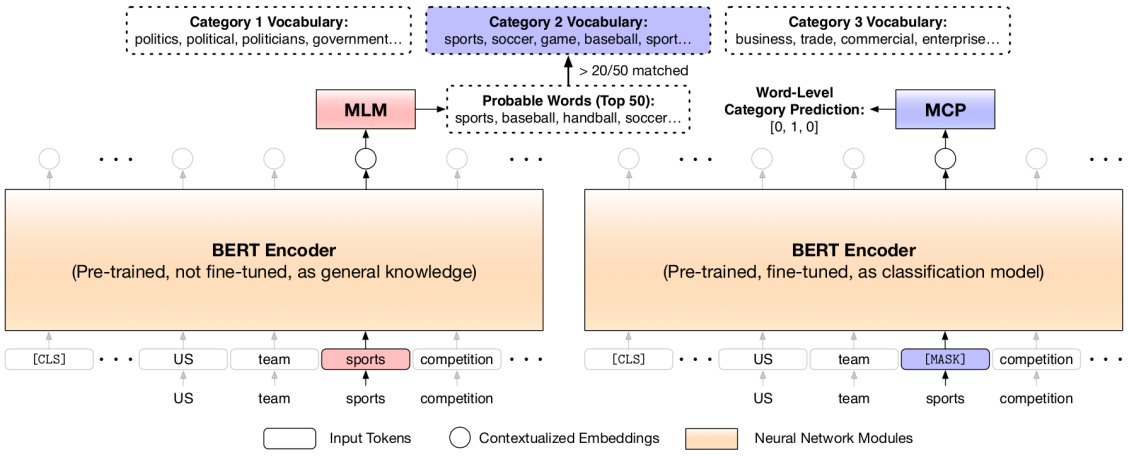

Figure 2: Overview of training step 1 (on the left) Masked Language Modeling and step 2, Masked Category Prediction (on the right). Source: [Meng et al., 2020, 6]

In a third and last step, the model is generalized on document level since the objective is to assign each document to exactly one category. This is also approached through self training of the pre-trained language model (but instead of self training on word level) on document level. Until now, not each document has been seen by the model, since not all documents contained category indicative keywords. Applying the model on document level further generalizes the model. The classifier of step two, that has been used to predict the category of a masked-out word, is now applied to the encoded [CLS] token. This embedding considers the whole sequence/document. In this step, KL-divergence loss is used to guide the model's classifier prediction towards the target distribution, which is obtained using a soft labeling strategy. The target distribution is derived by enhancing high confidence predictions and demoting low-confidence predictions using squaring and normalizing of current predictions [Xie et al., 2016; Meng et al., 2020].

3.3 Cluster Analysis

Clustering is a method of unsupervised machine learning, where data points are classified into meaningful groups based on similarities. Each data point is expressed through its features [Rodriguez et al., 2019]. In topic modeling, a data point could be a document whose features are the topic weights. I.e., documents that contain similar topics can be expected to be assigned to the same cluster (or topic in our example). The result is a plot, showing relative distances of the data points in the feature space, reduced to two dimensions. Often, the color of the assigned data points shows cluster assignment. Application of a dimensionality reduction algorithm like Principal Component Analysis (PCA) or t-distributed stochastic neighbor embedding (t-SNE) is often required due to limitations of the cluster analysis algorithm [van der Maaten and Hinton, 2008].

Various cluster analysis algorithms are available, K-Means is probably the most popular, but the number of clusters needs to be known in advance which can be disadvantageous. Then, there are density-based approaches (like mean-shift clustering, or HDBSCAN) that discover the number of clusters automatically. Other approaches include hierarchical methods (e.g., Agglomerative Hierarchical Clustering) which produce in addition to the two dimensional plot a dendrogram that represents cluster hierarchy [Rodriguez et al., 2019; Bano and Khan, 2018].

4 Datasets

This chapter will provide an overview over all datasets used for the evaluations in the following chapters. Further, we will discuss differences and suitability of the different datasets for the task of topic modeling and document classification.

4.1 Dataset Overview

Name	#Topics	#Train Documents	#Test Documents	Labels
20Newsgroups	20	11'314	7'532	available
AG's News	4	120'000	7'600	available
W1	not defined	20'000	300	not available
W2	not defined	149'700	300	not available

Table 1: Overview of the datasets used for evaluation. Datasets W1 and W2 do not have a predefined number of topics, in the corresponding paper Bianchi et al. [2021b] values 25, 50, and 100 were used.

4.2 20Newsgroups Dataset

This labeled dataset is probably the most widely used dataset in multiple domains of natural language processing. It is integrated into various machine learning tools and frameworks such as the python-package *scikit-learn*. It is widely used for tasks such as topic modeling, document classification and clustering. The amount of texts contained are 18'000 newsgroups posts split evenly into 20 topics[1], some are strongly related [Mitchell, 1997]. Sometimes the differences between topics are rather subtle

[1] 'alt.atheism', 'comp.graphics', 'comp.os.ms-windows.misc', 'comp.sys.ibm.pc.hardware', 'comp.sys.mac.hardware', 'comp.windows.x', 'misc.forsale', 'rec.autos', 'rec.motorcycles', 'rec.sport.baseball', 'rec.sport.hockey', 'sci.crypt', 'sci.electronics', 'sci.med', 'sci.space', 'soc.religion.christian', 'talk.politics.guns', 'talk.politics.mideast', 'talk.politics.misc', 'talk.religion.misc'

such as the hardware topics, which distinguish manufacturers (mac, windows and IBM).

We obtained the *bydate version* of the dataset via *scikit-learn* and removed headers during fetching, as well as email addresses and meta-data hinting at the category/newsroom.

4.3 W1 and W2

The datasets W1 and W2 were obtained from the *DBpedia* abstract corpus[2] that contains the text of the first introductory section of the articles. According to Wikipedia guidelines, the first sentence of the introduction is used to situate the article into its broader context - an interesting property for topic modeling as it ensures that related concepts are mentioned. Links that these abstracts contained were extracted separately into an unordered set. The abstract text itself contains no links. Thus, links that could include hints for the model as to what topic is referred are omitted. Abstracts are available in 7 languages [Brümmer et al., 2016; Bianchi et al., 2021b].

The authors of CTM, Bianchi et al., collected 100'000 random abstracts from the English DBpedia abstract corpus - this dataset is being referred to as W2. 300 of the articles from W2, are used as test set. The corresponding 300 articles in the other test languages, German, French, Italian, and Portuguese constitute the parallel test sets. W1 is a smaller version of W2 that was used to evaluate performance of CTM to baseline topic models in a standard (monolingual) setting. Test documents are the same as in W2. Both datasets W1 and W2 are unlabeled, and the number of topics is unknown.

Lastly, it needs to be noted that the test documents are all referring to the same entity, but they were written by different authors independently in each language. Thus, there is not an exact semantic correspondence between them. Emphasis can be set on different aspects, and some abstracts are minimalistic while the corresponding article in another language is exhaustively detailed. Consequently, topic weights can differ between the abstracts of different languages, even though their general entities correspond.

[2]see: `https://downloads.dbpedia.org/2015-04/ext/nlp/abstracts`

4.4 AG's News

AG's News is a collection of more than 1 million news articles from 2'000 sources that are gathered by an academic news search engine. The collection dates back to the year 2004[3]. Zhang et al. extracted from this corpus a benchmark dataset for text classification consisting of news articles obtained from the 4 largest classes - "World", "Sports", "Business", and "Sci/Tech". In total, AG's News consists of 30'000 train articles and 1'900 test articles per class [Zhang et al., 2015]. Meng et al. [2020] used the same benchmark dataset. The number of training documents is not differing much in comparison to W2, although the test set of AG's News is approximately 25 times larger.

[3]see: `http://groups.di.unipi.it/~gulli/AG_corpus_of_news_articles.html`

5 Evaluation Techniques

In this chapter we explain how we approach evaluation of the two approaches in cross-lingual topic modeling. We will compare a weakly-supervised method, and an unsupervised method and evaluate their capabilities to predict topics of unseen documents and languages. As there are no labels available on the main dataset, W2, evaluation is posing a challenge.

5.1 Monolingual Evaluation

We will first evaluate performance of the two baseline models in a monolingual setting. Both baseline models use embeddings obtained by the pre-trained model *bert-base-uncased*. To evaluate topic quality the measure NPMI coherence is chosen. Especially, we want to obtain a comparison between the obtained topic's most important words.

5.1.1 Metrics of Monolingual Evaluation

NPMI-Coherence is a measure that automatically evaluates coherence of topics based on the most probable words per topic. Larger values hint at a better topic model that produces more coherent and interpretable topics [Rosner et al., 2014]. An implementation of the topic modeling package Gensim is used (*c_npmi* coherence model[1]) that follows Röder et al. [2015]. The approach we use is based on a sliding window and the normalized pointwise mutual information (NPMI) of all word pairs of the 10 most important words per topic. In order to measure similarity between words, a Wikipedia corpus is used as an external reference to create a semantic space and embed the words into it. The authors weight the vectors obtained via co-occurrence counts by normalized pointwise mutual information[2] and calculate then

[1] `https://radimrehurek.com/gensim/models/coherencemodel.html`

[2] see: Aletras and Stevenson [2013] for detailed explanations.

the cosine similarity between any two word representations [Aletras and Stevenson, 2013].

Accuracy This measure is a comparison of how many predictions match the truth labels. We use the topic that has the most weight in each document as category assignment for that document. It is calculated by dividing the sum of matching predictions by the total number of predictions. Accuracy can only be computed when trustworthy labels are available, and in the setting of topic models, when the model can follow the same scheme of categorization, as is possible with topic models that receive supervision input. This ensures that predicted category and truth labels correspond to each other. Reassigning topic IDs for the prediction until the best accuracy is reached can improve accuracy, but only in cases where the categorization schemes match. In other words, the underlying topics that were chosen must also be the topics that the model is computing. In the case of unsupervised topic models, the schemes do not overlap. In this case we calculated Pearson's correlation coefficient between the labels ID and the predicted topic ID and evaluated whether there were hints of correlation. But we could only find anti-correlation, thus we will not further evaluate this approach. W2 has not been labeled, and challenges arise, since categorization by topic is not a task that is unambiguous.

5.2 Visualizing Topic Models

Interpretation of a topic model can answer questions about the meaning and content of each topic, the prevalence of topics within the document collection or within a certain document. Also, relations between topics and between documents are explored.
To facilitate the interpretation of topic models, the two different underlying distributions (document-topic-distribution and term-topic-distribution) of a topic model are used in visualization tools such as *pyLDAvis*. A tool from Sievert and Shirley [2014] which originated as an R-package *LDAvis* and was adapted for Python compatibility. This interactive tool allows to visualize in one graphic, both distributions: document-topic-distribution ($P(T|D)$) and the distribution of terms within a given topic.

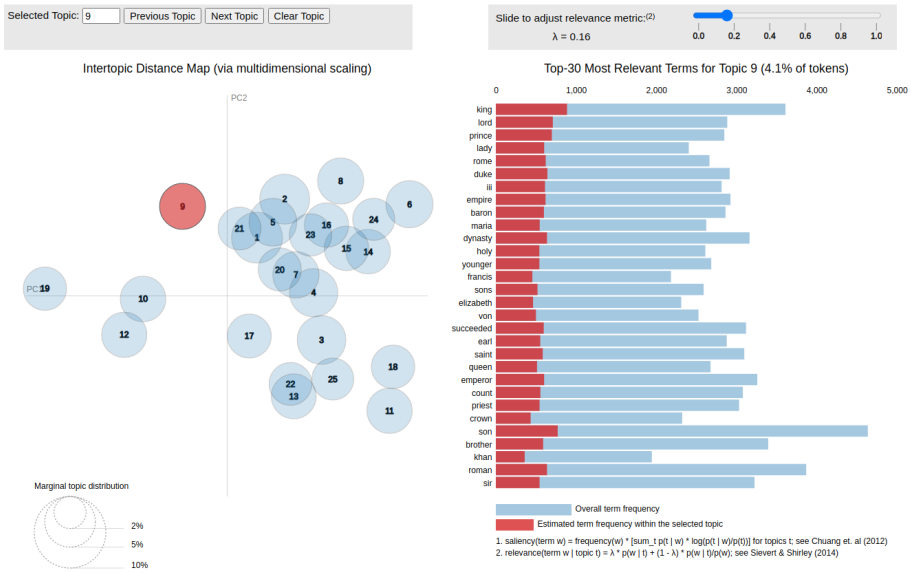

Figure 3: Example of a visualization of a topic model trained on W2 dataset. On the right side denotes the most important terms in a topic that seems strongly related to aristocracy.

Distinctiveness and saliency are metrics that measure how much information a specific term is conveying about a topic in a collection of documents. They can be found in the bar plot on the right part of the visualization. The distinctiveness of a term w computes the KL-divergence between the distribution of topics, given a term and the marginal distribution of topics [Chuang et al., 2012]:

$$distinctiveness(w) = \sum_{T} P(T|w) \log \frac{P(T|w)}{P(T)}.$$

Saliency is calculated by weighting a term's distinctiveness by its global relative frequency $P(w)$ [Chuang et al., 2012]:

$$saliency(w) = P(w) \cdot distinctiveness(w).$$

Larger values indicate that a term is more useful to identify a topic in the document collection. Words that have high saliency values, are most informative to identify topics. In the bar chart, both the saliency and the total term frequency are shown. Near exclusivity of a term to a topic is present, when $saliency(w) \geq P(w)$.

$$relevance(w|t) = \lambda \cdot P(w|t) + (1 - \lambda) \cdot \frac{P(w|t)}{P(w)}$$

The weight parameter λ is a relative weight parameter that defines which summand is used in the relevance metric. A λ value of 1 is regarding only the first summand, in consequence terms are highlighted that occur more frequently but might not be exclusive to the topic. Lambda values close to 0 are only regarding the second summand and thus highlight potentially rare, but more exclusive terms to a topic. The transition is continuous. A user study by Sievert and Shirley found that $\lambda = 0.6$ is best suited to interpret topics. This is expected to vary depending on the data and the specifics of topics.

The positions of topic circles are determined by the term-topic-distribution: A topic is represented by the importance of each word in the vocabulary of the topic. We can embed topics as points in N dimensions - with N corresponding to the number of distinct terms in the vocabulary. Using a dimensionality reduction algorithm, the representations of a topic will be reduced to two dimensions while preserving relative distances to other topics. Thus, the more words two distinct topics share, the closer the topics will be situated on the distance map. In case of pyLDAvis' distance map, the positions in the two-dimensional plot are obtained using the dimensionality reduction algorithm multidimensional scaling (MDS) [Tzeng et al., 2008].
The area of a topic bubble represents the number of words in the vocabulary that is assigned to the topic [Sievert and Shirley, 2014].
The CTM model, is based on an LDA topic model, which allows us to use this pre-build visualization feature in order to determine the quality of topics both in a quantitative and qualitative way.

5.3 Multilingual Evaluation

We follow Bianchi et al.'s approach and evaluate the transferred topic model on parallel test sets in multiple languages. Quintessential is the availability of multilingual pre-trained embeddings in the languages the test sets will be in, otherwise, the transfer would not be straightforward. We then compute evaluation metrics "Matches", "Centroid Similarity" and "KL-Divergence" between the test set in the original language and a parallel test set in another unseen language, where the model has been transferred to. The test sets are semantically parallel in a sense that corresponding abstracts cover the same entity or subject.

We hypothesize that different versions of pre-trained multilingual contextualized embeddings will result in improved results on the multilingual evaluation metrics. We use - in contrast to Bianchi et al. [2021b] - three different versions of pre-trained multilingual contextual language models. The first model, *bert-base-multilingual-*

uncased, is used with mean pooling to derive article representations. We also use this model without mean pooling in LOTClass. Furthermore, for CTM we use two other models: (1) The same model as in Bianchi et al. [2021b] *paraphrase-multilingual-mpnet-base-v2* which maps sentences & paragraphs to a 768 dimensional dense vector space. And (2) *distiluse-base-multilingual-cased-v1* which uses a lighter version of BERT.

5.3.1 Metrics of Multilingual Evaluation

As in Bianchi et al. [2021b] we evaluate whether transfer learning is applicable on the topic modeling task through use of multilingual embeddings to represent the input texts. Bianchi et al. used three evaluation metrics that account for totally and partially matching predictions between the test sets in English (the original training language of the model) and the unknown, foreign language. Note, that the foreign language is not unknown to the multilingual language model, since we use pre-trained versions of mBERT.

Matches are defined by the percentage of matching predictions per document between the English test set and the corresponding article in the unseen language's test set. Higher scores are better. The downside of this evaluation metric is that predictions are formed based on the maximum topic weight in each document, even if there is no clear dominant topic. This measure contains no certainty threshold to ensure only safe predictions are made.

Centroid Distance To account for similar predictions, the authors Bianchi et al. compute the centroid distance between the two differing topic predictions. They use static word embeddings derived from *word2vec* and calculate the average embedding of the top-5 most category representative words for both predictions [Mikolov et al., 2013]. In a second step the cosine similarity between the two centroid vectors is calculated. Cosine similarity between two vectors can be assigned all values in the interval from -1 to 1. The larger the value the closer are two topics related [Sitikhu et al., 2019].

Average KL divergence compares the distributional similarity between the two predicted topic distributions for each document pair on the test set. Lower is better, indicating that the two distributions do not differ much [Bianchi et al., 2021b]. This is a hint at consistent predictions, which also accounts for partial similarity.

6 Evaluating Training Dynamics of CTM and LOTClass

This chapter will in a first part evaluate - CTM and LOTClass - in a standard setting of topic modeling where the model is utilized to find the underlying tropes and topic mixtures contained in a large training corpus. In such a setting, without labeled data available, we will use unsupervised evaluation metrics such as the NPMI coherence measure, which quantifies the quality of a topic model by its derived most important words per topic.

In the second part we will use zero-shot learning to apply our trained models to cross-lingual test documents. The two models were trained using multilingual embeddings as input representations. Representing input documents by use of multilingual pre-trained embedding techniques allows the application of a trained topic model on another unseen language[1].

6.1 Evaluation in a Standard Setting

First, we will evaluate our two models in a standard, monolingual setting in order to verify that we can reproduce the original work's indicated performance. In Bianchi et al. [2021b] a CTM model was trained on the W1 dataset which consists of random sampled Wikipedia abstracts obtained by the authors. LOTClass [Meng et al., 2020] was evaluated originally on the AG's News dataset. However, both datasets are not commonly used in document classification tasks, let alone topic modeling.
In order to compare the two methods among each other, the 20Newsgroups dataset was chosen. This dataset is often utilized on the task of document classification and topic modeling. Problematic is the much smaller number of articles contained in

[1] All scripts and source code can be found here: `https://github.com/frmati/TM_Material`, additionally, the output files of the trained models are available here: `https://drive.google.com/drive/folders/1mFSHXuTYWw8qrlIT8WSuH_T538PA0bca?usp=sharing`.

that dataset. Another difference is the number of topics that needs to be determined in advance for both models (4 topics in AG's News, and 20 topics in 20Newsgroups). As in Bianchi et al. [2021b] we set the number of topics on the W1 and W2 datasets to 25. For the sake of consistency, we will always use the datasets W1 and W2 with 25 topics. Very similar results were obtained by Bianchi et al. [2021b] who used 50 topics in the monolingual evaluation setting[2].
In Table 2, we present an overview of CTM and LOTClass trained under standard configurations on the datasets AG's News, W1, W2 and 20Newsgroups.

Dataset	#Topics	Method	NPMI-Coherence	Accuracy
AG's News	4	LOTClass	-0.003	0.854
AG's News	4	CTM	0.010	-
20Newsgroups	20	LOTClass	-0.074	0.078
20Newsgroups	20	CTM	0.018	-
W1	25	LOTClass	0.03	-
W1	25	CTM	0.177	-
W2	25	LOTClass	0.137	-
W2	25	CTM	0.197	-

Table 2: Topic evaluation metrics on the various datasets are calculated based on one run. Thus, some variance in the results is to be expected. On the unlabeled datasets, accuracy calculation is not possible. Due to the unsupervised approach of CTM, we could not calculate accuracy of the CTM model on 20Newsgroups.

Accuracy calculation is not possible on the datasets W1 and W2, as they were not annotated. For the unsupervised model, CTM, no meaningful accuracy results could be obtained, since the model's scheme of classification does not correspond to the dataset's labels, as we cannot ensure that an unsupervised topic model utilizes the same annotation scheme as the gold classification labels. To address these limitations, we (1) calculated alternative measures such as Pearson's correlation coefficient between the labels and the predictions and (2) reassigned topic IDs of the predictions (since they can be chosen arbitrary). Both approaches were unsuccessful, and it needs to be noted that the second approach becomes increasingly expensive the more topics the dataset contains.
LOTClass on the other hand is a weakly-supervised method, which allows to make sure the annotation schemes of model and labels match. We were able to reproduce the accuracy of $\approx 86\%$ on AG's News. Application of the model on the 20News-

[2]Evaluations of Bianchi et al. [2021b] resulted in NPMI coherence values of 0.1658. This value was obtained by using top-10 words per topic on the W1 dataset. In contrast to us, [Bianchi et al., 2021b] set the number of topics to 50 on the W1 dataset.

groups dataset, which in comparison to AG's News contains 10 times less training material, and 5 times more topics, showed the limitations of LOTClass' approach. Based on the supervision input labels, each category's vocabulary size was extended to 50 by adding words that occurred in similar contexts using the masked language modeling objective. But many category words did not occur in enough documents to provide fine-tuning guidance. Also, in the second step of training, the number of words judged as category-indicative was too small to obtain accurate predictions on the document classification task.

This problem can also persist on larger datasets, such as the W2, where not enough category representative articles were found using the standard threshold for category indicativeness (of a word[3]). In Meng et al.'s default configuration that was provided for AG's News dataset, this value was set to 20/100. In order to improve the model, we tried to lower this threshold but without improvement. Another approach could be to change the supervision input. Instead of using the top words per topic from the CTM model, another categorization scheme could be used, that was defined manually.

For further training of LOTClass we leave the threshold value of step two set on the recommended value and continue training the model on document level (step 3 in Chapter 3.2.4.1). NPMI coherence of LOTClass is comparable to CTM on the W2 dataset.

6.1.1 Influence of k on NPMI Coherence

NPMI coherence scores are often used in literature to automatically estimate a topic model's quality. In the context of NPMI coherence, the variable k denotes how many category representative terms are being used to calculate the coherence score of a topic model. Choosing k smaller will consider only the most coherent words in the calculation, which will influence NPMI score positively, as can be seen in Figure 4. We evaluate the effect of k on the derived topic coherence in order to estimate the magnitude of change. In Bianchi et al. [2021b], k was chosen to be 10. We train and evaluate a CTM model on W2 and vary k from 2 to 20. It becomes obvious that, since no guidelines on the size of k are available, NPMI scores can easily be influenced in a favorable direction. When comparing different models based on NPMI coherence scores, one needs to make sure to consider the exact configurations

[3]See Section 3.2.4 for a detailed description

used during their calculation. We note that in our setting choosing $k = 10$ vs. $k = 20$ increases NPMI coherence score by 0.03.

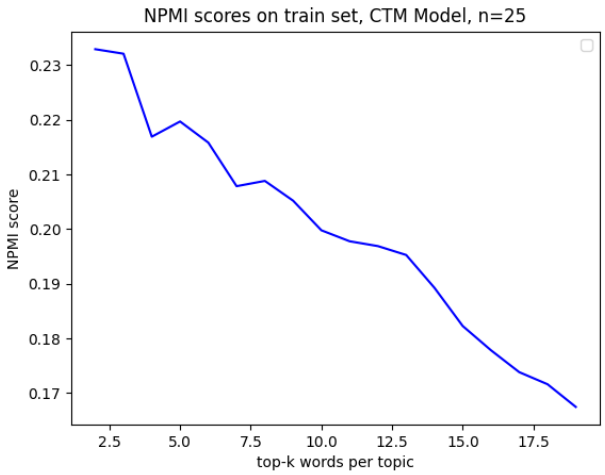

Figure 4: NPMI values in function to k (top-k words per topic considered during calculation). The CTM model was trained with $n = 25$ topics. Coherence values can only be calculated for discrete values of k.

6.1.2 Influence of Training Repetitions on Topics Derived

We train a CTM model with default parameters using *paraphrase-multilingual-mpnet-base-v2*'s sentence representations on W2's training set and evaluate the resulting topics with pyLDAvis [Sievert and Shirley, 2014][4]. Increasing the number of training epochs leads first to better separated general topics and a rapid decrease of loss (Figure 5). Between 200-400 epochs, train loss is only changing slightly and the topic representations of pyLDAvis obtained via multidimensional scaling are starting to overlap (Figure 8) until we reach a point where only few topics are visible (Figure 9). This process continues until the most important words are associated to one topic. We will use pyLDAvis to show this process graphically. Note that topic IDs and positions can change due to involvement of random processes. In the illustrations included here, we focus on the topic containing words related to the Olympic Games[5].

[4] An overview over the default model configurations can be found in Section B.

[5] Interactive versions of topic overviews can be downloaded from: `https://github.com/frmati/TM_Material/tree/main/Training%20Process`

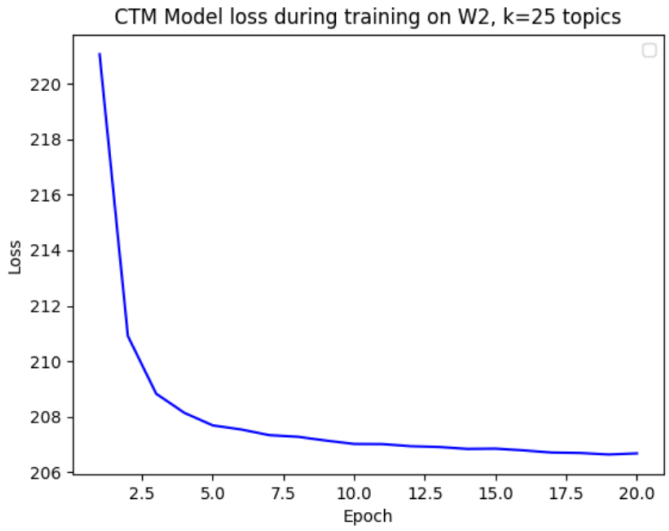

Figure 5: CTM model train loss decreasing rapidly during the first 3 epochs.

After one epoch of training, the estimated term frequency within each topic is set to approximately the same value for every word (Figure 6). All estimated frequencies are significantly smaller than overall term frequencies, thus, words do not belong exclusively to a topic, but can rather be present in multiple topics. Already now the most frequent words per topic are suited to interpret the content of most topics. This is due to the information contained in the contextualized embeddings of the input documents that provide a first semantic segmentation [Bianchi et al., 2021b,a].

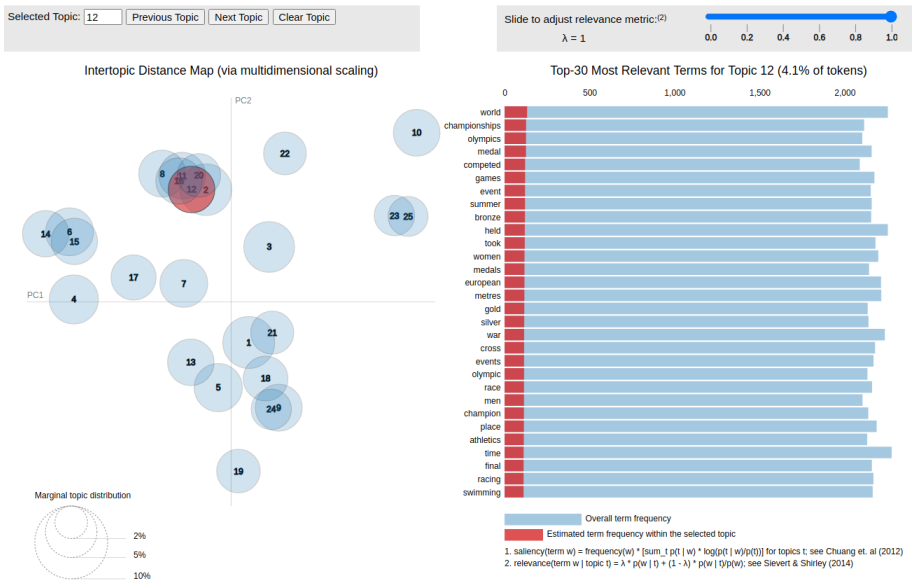

Figure 6: The Olympic topic is already noticeable after 1 epoch of training. Estimated term frequency within each topic is set to approximately the same value for every word.

By setting $\lambda = 1$ in the relevance formula, we consider only the conditional frequency of a word w occurring in a certain topic t ($P(w|t)$)[6]. Continuing training of the CTM model results in increasingly larger relevance values. Thus, as fewer words contribute heavily to a topic, topics become more specialized.

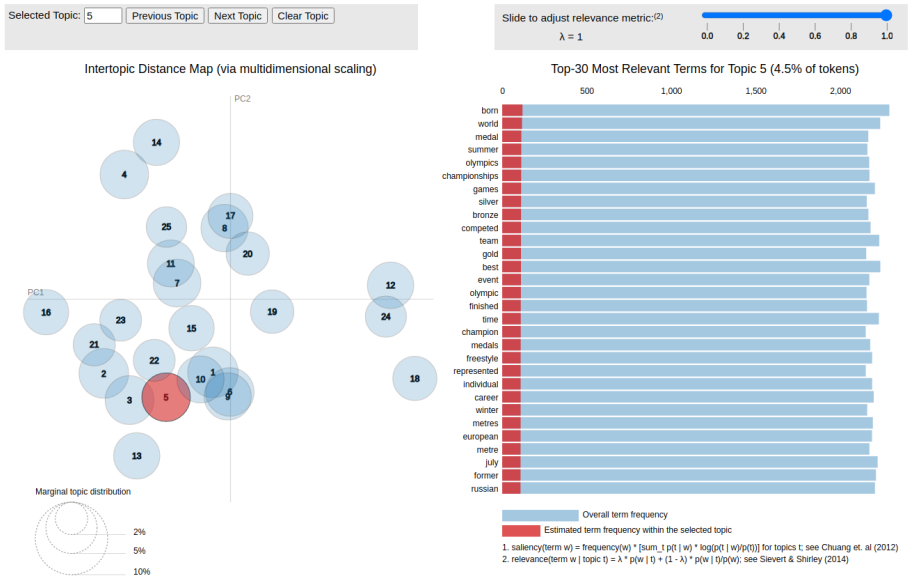

Figure 7: After training during 20 epochs, topics become better separated from each other.

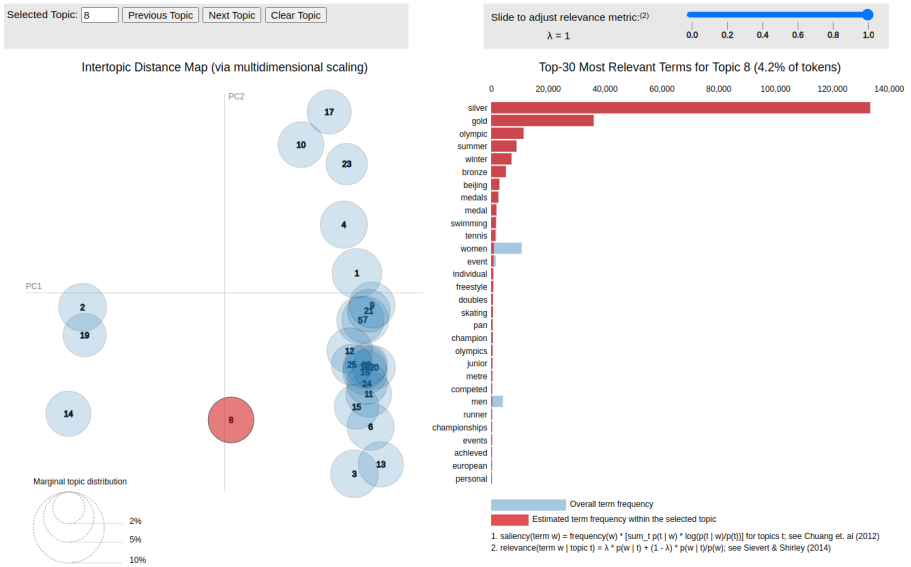

Figure 8: After training during 400 epochs, topic circles start to converge.

[6]more details available in Section 5.2.

On pyLDAvis' plot of the inter-topic distance map, that was obtained via multidimensional scaling, we see that the topics converge. In extreme cases, when training during 1000 epochs, we obtain approximately eight visible clusters (Figure 9).
The multidimensional scaling algorithm is comparing all topic-word distributions and projecting these distributions to a two-dimensional representation. Thus, relative distances between topics containing similar word occurrences will be smaller and topics that contain largely diverging topic-word distributions will be distanced from each other. In cases where the same words contribute to a topic, the representations overlap. The more similar term-topic distributions are, the closer will two topic circles be in the two-dimensional plot. We observe that words belong almost exclusively to only one topic, and thus the topic representations should be well separated. This phenomenon needs to be further evaluated especially by investigating the limitations of the multidimensional scaling algorithm regarding sparse topic-term-matrices [Tzeng et al., 2008].

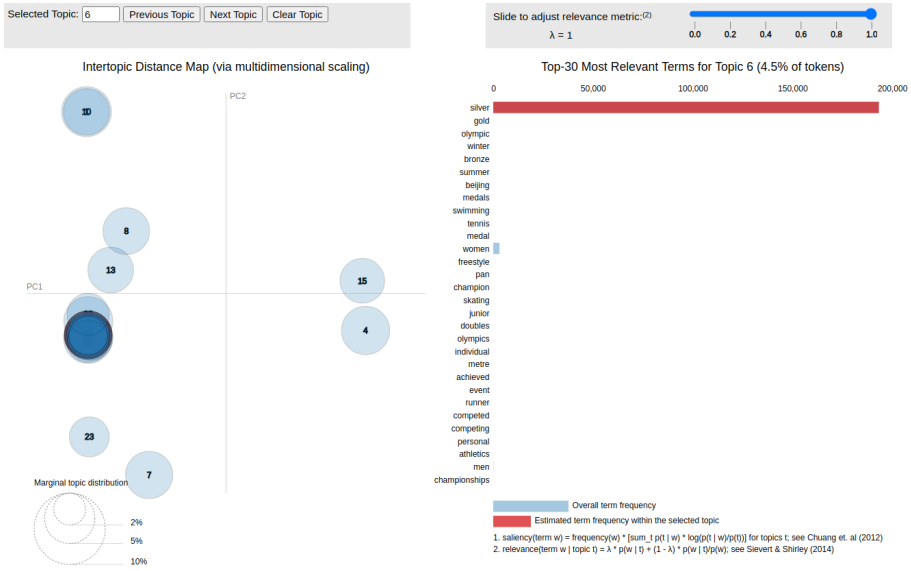

Figure 9: Training during 1000 epochs lead to extreme convergence of the topic representations.

6.1.3 Influence of Learning Rate on Loss

CTM is based on autoencoding variational Bayes, and an inference method adapted to be compatible with LDA - Autoencoded Variational Inference for Topic Models (AVITM) [Srivastava and Sutton, 2017]. The encoder model uses the observations to infer the hidden variables through use of an approximated learned distribution which makes inference traceable. The utilized loss function incorporates two ele-

ments: (1) the divergence between the learned approximated distribution and the true prior that is assumed to be an approximation to Dirichlet distribution for each dimension of the latent space and (2) the reconstruction likelihood [Srivastava and Sutton, 2017; Kingma and Welling, 2019].

In order to eliminate the possibility that the model, while trying to reduce loss, reached a local minimum and thus was not able to find the global minimum of the loss function, we trained the model with learning rates varying from 1×10^{-10} to 1. For learning rates larger than 0.2, the training process quit early, since loss could not be decreased.

It seems that the approximation to the assumed true prior distribution (a Dirichlet distribution) is learned quite fast. The same holds true for the minimization of reconstruction error, in other words: The maximization of reconstruction likelihood. Changing learning rates manually had no effect on train loss reduction overall. It affected only the number of epochs that were needed to reach a minimum amount of loss. Hence we continue to use the default learning rate of 0.002 [Bianchi et al., 2021b]. In Section A.0.1 an overview over the topics derived using CTM and LOTClass can be found.

6.2 Evaluation in Cross-lingual Setting

We keep training the CTM model on the full W2 dataset in English and compare topic predictions between the abstracts of the English test set and the respective counterparts in French, Italian, Portuguese, and German.

6.2.1 Evaluation on a Parallel Test Set

The test sets of the W2 dataset are covering the same entity in a general sense, but the articles were written by various authors independently, which leads to differences in emphasis of certain aspects and to differences in terms of content. The prediction of a perfect topic model for a given article would not be influenced by such differences since the most dominant topic would still be related to the abstract's title. In reality there are differences in the similarity metrics when comparing the derived English topics with the topics from the same document in e.g., German. These differences could be caused by many reasons. Some of the reasons will be explained in the following sections.

We want to derive the same evaluation metrics on a dataset that is as far as possible equal in terms of content. Therefore we automatically translate the German test set into English[7], and compare the predicted topic weights between the original German test set with the translated test set. This allows us to eliminate most of the differences between two test sets and make them more comparable such that we can calculate the multilingual evaluation metrics in the best possible setting. Automatic translation can also be problematic in a sense that we now must be aware of biases induced by automatic translation, also semantic ambiguities can sometimes be challenging for neural translation systems [Koehn, 2020, 5-8]. Thus, automatic translation cannot guarantee that the two test documents are exactly similar in terms of their content. When we have as much semantic similarity as obtainable using automatic translation models, the matching topic predictions are 8 percent points higher than on the other test sets that were authored independently of each other (results can be found in Table 6).
Quintessentially we observe, that even under the best possible conditions, we cannot get a match score of 100% and thus matching predictions of topics for all document pairs. The topic model is transferable, but with limitations, partly caused by the multilingual document representations.

6.2.2 Influence of Training Duration on Multilingual Evaluation Metrics

After approximately 4-5 epochs, loss decreases only marginally (Figure 5). This holds true even for an unrealistically large number of epochs (see Figure Figure 15 in the Appendix). We compare the influence of training epochs on the results of the multilingual evaluation metrics.

Both vocabulary distribution within topics, and the distribution of vocabulary between topics changed notably during the training process. These distributions are calculated based on observations on the train set.
In terms of the predictions on the different test sets, interestingly, no effects on the multilingual evaluation metrics were observable. The increased specialization of topics, that attributed large expected term frequencies within the topic to the most important keywords, seem not to have an effect on the test set prediction of topic weights. We assume that occurrence of important topic words in the test set

[7] using DeepL's free API: `https://www.deepl.com/en/docs-api/`

are still the main factor when deriving topic weights. As we have investigated, the most important words stay approximately the same during the training process, and the calculated relevance value does not affect classification. Thus, we conclude that the number of epochs does not seem to influence the results of the multilingual evaluation metrics. We still obtain similar results.

Epochs	Ø Matches (in %)	Ø Centroid Distance (n=25)	Ø KL Divergence	NPMI score
20	72	0.81	0.28	0.2
100	72	0.80	0.32	0.20
300	71	0.79	0.33	0.19
500	67	0.78	0.36	0.18

Table 3: Varying the number of training epochs of the CTM model, does not affect the metrics Matches, Centroid Distance and KL Divergence. Ø denotes average values between the test sets in French, German, Italian, and Portuguese. Detailed results can be found in Section A.0.5.

6.2.3 Influence of Embedding Models on Multilingual Evaluation Metrics

We evaluate the effects of using different models of pre-trained multilingual text representations. The mBERT model *mBERT-base-uncased* determines our baseline, as it is not fine-tuned for the task of producing sentence embeddings. It is usually used for the task of masked word prediction. However, we can use mean-pooling, which produces an embedding vector that corresponds to the average overall token's embeddings occurring in the sentence [Mohebbi et al., 2021].
The models *paraphrase-multilingual-mpnet-base-v2* and *distiluse-base-multilingual-cased-v1* are pre-trained multilingual sentence transformers, that are suited for sentence-similarity calculation [Reimers and Gurevych, 2019]. The two sentence transformers can discriminate lowercase from uppercase letters, while *mBERT-base-uncased* is not case sensitive. Preprocessing of the input text is adapted accordingly.

Paraphrase-multilingual-mpnet-base-v2 is performing best on the multilingual evaluation metrics. However, there is no notable difference in terms of the quality/coherence of topics derived from the train set. As expected, mBERT in combination with mean pooling performed worse, due to not being fine-tuned on the task of sentence similarity prediction.

Embedding Model	Ø Matches (%)	Ø Centroid Similarity	Ø KL Div.	NPMI
mBERT-base-uncased mean-pooling	59	0.72	0.44	0.20
paraphrase-multilingual-mpnet-base-v2	**72**	**0.81**	**0.28**	0.20
distiluse-base-multilingual-cased-v1	69	0.78	0.33	0.21

Table 4: Average CTM results (denoted by Ø) between the German, French, Italian and Portuguese test sets in comparison to the English test set. All results based on 20 epochs of training, NPMI score calculated on 20 words per topic.

Model	Parameters
bert-base-multilingual-uncased	mBERT, 102 languages, embedding dim: 768, case-insensitive 30'000 subwords, max. sequence length: 512
paraphrase-multilingual-mpnet-base-v2	XLM-RoBERTa, 100 languages, embedding dim: 768, case sensitive 50'000 byte subwords, max. sequence length: 128
distiluse-base-multilingual-cased-v1	DistilBert, 15 languages, embedding dim: 512, case sensitive 30'000 subwords, max. sequence length: 128

Table 5: Properties of the different contextualized embedding models used. All models use mean-pooling over tokens to derive an embedding of the input sequence [Devlin et al., 2019; Liu et al., 2019; Reimers and Gurevych, 2020; Sanh et al., 2019].

6.2.4 Multilingual Evaluation Results of CTM

We use the CTM model with configurations that worked best during the experiments of the sections before. This involves the choice of embedding model *paraphrase-multilingual-mpnet-base-v2*, the number of training iterations (20), and learning rate (0.002). We still use the W2 dataset and mine 25 topics.

The results of Bianchi et al. [2021b] on the W2 dataset with 25 topic were reproduceable. Evaluation results will suffer when the number of topics is increased. Especially the number of matches will decrease, as the match metric does not account for partial correctness and matches are obtained using the largest topic weight, no matter how small the difference between the two most dominant topics might be. Metrics that do account for partial correctness - such as Centroid Distance or KL Divergence - seem to be more consistent when increasing the number of topics.

	Matches (in %)	Centroid Distance	KL Divergence
Italian	74	0.82	0.23
Portuguese	75	0.82	0.27
German	71	0.80	0.30
French	72	0.80	0.31
EN (translated)	80	0.87	0.21

Table 6: Comparison of the multilingual evaluation metrics for each test set language separately in comparison to the English Test set. Last row is the comparison of a new test set, the automatic translation from German into English, with the German test set.

We visualize the prediction results of CTM on the German test set (Figure 10). The equivalent visualization on the English test documents can be found in Figure 16[8].

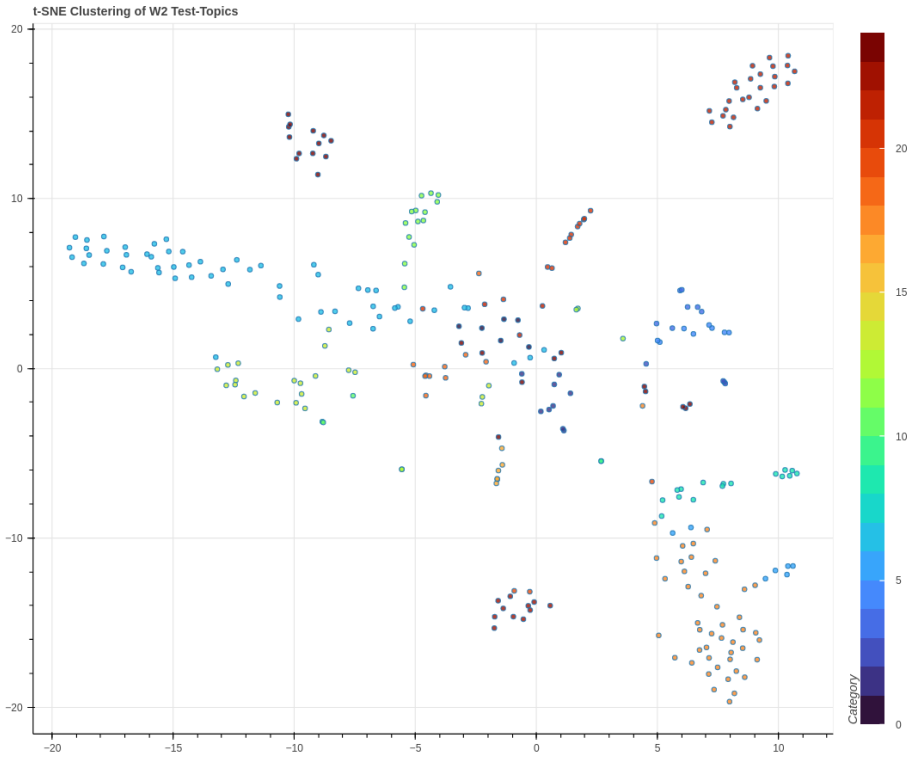

Figure 10: CTM's prediction of topic weights on the German test documents. Colors represent the most dominant topic, positions are obtained using t-SNE dimensionality reduction, therefore positions are to be interpreted relatively.

[8]Interactive versions of the graphics in HTML-format can be found here: `https://github.com/frmati/TM_Material/tree/main/Test%20Documents`

Each data point represents a test article that is characterized through its predicted topic-weights. A point resides thus in a feature space of 25 dimensions (25 topics). To visualize the relative distances between data points and as consequence their similarities in terms of topic content, we reduced the number of feature dimensions from 25 to 2 using the dimensionality reduction algorithm t-SNE. Positions and distances to other data points need to be interpreted relatively. A data point's color represents the topic ID that was attributed the most weight in the document.

In Figure 10 we see some clearly defined clusters of documents that contain very similar topics and are clearly separated from other abstracts. In visualization's center, we cannot find dense clusters with a common topic prediction. These documents have similarities in terms of their topic weights.

6.2.4.1 Manual Evaluation of Topic Predictions

Non-matching topic predictions will be evaluated between the test sets in English and German. It needs to be noted that the text of two articles is not a translation, but rather two articles that originated independently in the two languages by different authors, with only the general subject in common. Thus, there can be a very different emphasis on certain details. The maximum length per abstract was limited to 200 tokens by Bianchi et al., due to limitations of the sentence embedding method.

Possible reasons for non-matching predictions are that two articles have differing lengths. In these cases, the focus of content often lies differently, and more emphasis is put onto other subjects. The shorter document is more often classified correctly in cases where the predictions do not match. These abstracts seem to focus more on the key topic.
In many cases the two predictions are partially correct, as the two topics are related and an abstract exhibits content of both - i.e. CTM topic ID 14 (car/tennis championships) and ID 19 (tournaments, team championships)[8].
Furthermore, compounded German nouns seem to make correct topic predictions more difficult. In such cases, topic informative words are not correctly recognized and the hints to the correct category seem to vanish.

[8]More details can be found in the appendix (Table 8).

6.2.5 Multilingual Evaluation Results for LOTClass

Until now, we have been focusing on CTM's performance on the W2 dataset. We will now examine LOTClass' results in the multilingual setting of the W2 dataset. In order to make both CTM and LOTClass better comparable, we use the top-k most representative words per category obtained by CTM as weak supervision input for LOTClass. Label names occurring in multiple categories were removed manually. The resulting category vocabulary after training can be found in Table A.0.2. *Bert-base-multilingual* was used as a pre-trained language model to allow for evaluation of the trained model on cross-lingual test data.
In the following section, we will evaluate the effect of the number of labels per category that are provided to the model as weak-supervision input and calculate multilingual evaluation metrics.
The approach of LOTClass is to fine-tune a BERT model to understand categories [Meng et al., 2020]. But on the W2 dataset this approach was not very successful, as the model was not able to find enough category representative terms per document. We can therefore not evaluate a model that was trained under the best possible conditions.

6.2.5.1 Effect of Weak-Supervision Input on Multilingual Evaluation Metrics

We use as weak supervision input the top-k most representative words per topic, as they were derived using the CTM and train LOTClass with $k = 1, 5, 20$ words per topic. In cases where labels names occurred in multiple categories, they were removed, resulting in a smaller amount of supervision input for certain categories[9].

k	Ø Matches (%)	Ø Centroid Distance	Ø KL Divergence
1	67	0.76	0.94
5	**86**	**0.88**	**0.44**
20	81	0.84	0.53

Table 7: Multilingual evaluation metric obtained using LOTClass on W2 dataset with 25 topics, we vary k (number of labels as weak-supervision input per category). Ø denotes average values between the test sets in German, French, Italian and Portuguese

Using $k = 5$ resulted in the best performance concerning the multilingual evaluation metrics (see Section A.0.6). One label per topic was less suited, as even fewer cat-

[9]The labels used as weak-supervision input to LOTClass can be found in Table 9 in the appendix.

egory indicative words could be found in this setting (see Table 12). Performance decreased with $k = 20$ as we used the most representative words per topic from the CTM model (see Table 11). This is problematic for two reasons: (1) the most category representative words are valuable inputs, as they belong nearly exclusively to a category. But the more of these top-k words we use, we include words as supervision input that are not category exclusive. The predefined categories will be defined less concisely and (2) the model gets computationally more expensive.

LOTClass' predictions of the most dominant topic per document are more similar between languages than the predictions of CTM. Also, LOTClass is outperforming CTM in terms of centroid similarities. These are larger for LOTClass, presuming that the two predictions for a parallel test document are more similar semantically. In contrast, KL divergence is larger for LOTClass than for CTM. Thus, the predicted document-topic distributions exhibit greater average difference for each test document.

Nevertheless, we need to be careful interpreting these results, as they could not be obtained under the best conditions. Under any configuration, LOTClass was not able to find the necessary amount of documents per category. According to Meng et al. [2020] this amount is set to 10 documents per category. Therefore at least 10 documents containing words considered category indicative need to be found. Then again, category indicativeness is based on a threshold of 20 tokens. Reducing that threshold is not recommended and will make the model less accurate because at some point each word will be considered category indicative. This will prolong training time.

As for some categories of the W2 dataset, no examples could be found, we did not change weak-supervision category labels, and did not reduce the threshold for category indicativeness. We continued the training process despite the warnings onto the 2nd and 3rd step. We note that the multilingual evaluation metrics (especially Table 10) without annotation labels are not reliable in that scenario, and we need to look further into the individual cases. The already familiar Figure 10 is helpful for that evaluation.

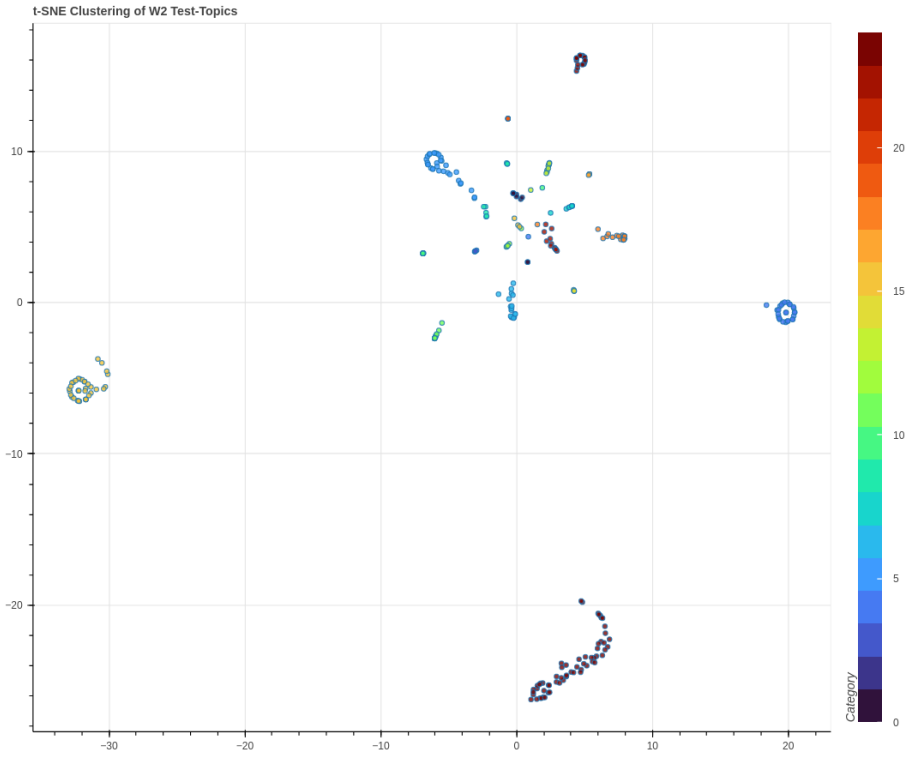

Figure 11: LOTClass' predictions of topic weights on the German test documents. Colors represent the most dominant topic, positions are obtained using t-SNE dimensionality reduction, therefore positions are to be interpreted relatively.

We can see that in comparison to the analogous classification of CTM's predictions in Figure 10 [10], LOTClass' predictions are visualized as more dense and clearly defined clusters. This is due to the incompletely trained model that could not find documents containing category representative terms for all available categories. Thus, these dimensions have weight 0 or cannot be recognized correctly during the prediction of topic weights on the test documents. Some abstracts have less topics in common and thus are more distanced from each other. On the other hand, if some terms occur in multiple documents, they are likely to be embedded very close to each other. Especially the articles contained in a circle formation are all equally similar to the article in the center of the circle as they share the most dominant topic [Wattenberg et al., 2016].

[10] The equivalent figure derived from the predictions on the English dataset can be found in Figure 16.

7 Conclusion

We followed Bianchi et al.'s approach and used their evaluation metrics that were designed to express how well a topic model could be transferred into a new unknown language. This language can be different from the language the model was trained on with the limitation that the two languages must be supported by the multilingual embedding model. Bianchi et al. [2021b] used the AVITM[1] framework as a basis, and changed the text input representation technique to contextualized pre-trained sentence embeddings. The availability of multilingual models allowed the application of a topic model in a cross-lingual way using zero-shot learning, which resulted in CTM. This model then was compared to LOTClass, an approach that is not using sentence embeddings but the fine-tuning of a BERT model for the task of category understanding.

Meng et al.'s results on the AG's News dataset were reproduceable, but we found that there were limitations on other datasets. This was the case on small datasets such as 20Newsgroups, but did occur on large datasets as well, if it did not contain enough documents with category indicative terms. On the basis of the keywords we provided as a starting point[2] the vocabulary was extended with semantically similar words. But these words did not occur in a sufficient amount of documents, which in consequence did not allow full adaption of the BERT model to the category prediction task. We could not train the LOTClass model under best conditions and must therefore evaluate the multilingual evaluation metrics obtained by LOTClass critically. It would be necessary to change the input labels to provide a better suited starting point to the model, on the other side this would make the comparison between the two models harder. Also, it would be beneficial to use another dataset that also includes a larger amount of test articles, since the W2 test sets only include 300 abstracts. The *XNLI* dataset could be adequate for that task, as it is also a standard dataset used for the evaluation of multilingual models and transfer learning. Furthermore, it provides test datasets in 15 languages that were manually

[1] Autoencoded Variational Inference For Topic Models, [Srivastava and Sutton, 2017]

[2] These keywords were obtained from CTM's most dominant words per topic.

translated by humans [Conneau et al., 2018]. But challenges remain, as there is also no gold categorization available.

We obtained another pair of parallel test datasets by automatically translating the German test set into English. This new pair should contain as much similarity in terms of contents as machine translation allows now. Nonetheless, we could never reach matching topic prediction using the CTM and obtain evaluation results that were larger than 80% resp. KL-divergence scores less than 0.20. Improvements in comparison to using abstracts in different languages of Wikipedia were noticeable. But there is still plenty of room for improvement, since still one fifth of the most dominant topic weight predictions do not match.

In the topic modeling task, which is mostly unsupervised, calculation of a metric reflecting accuracy is difficult due to non-matching topic schemes. Both could be correct and this needs to be manually evaluated. Therefore, topic models are not judged based on their predictions but based on the coherency of the most important words per topic. Evaluation in the case of LOTClass (which is a weakly-supervised classification model) is easier as the supervision input fits the annotation scheme of the labels.
We tested whether correlation metrics could be used as an accuracy measure for unsupervised topic classification, but this was not the case in our experiments. However, we conclude that visualizations of the topics and their most important words helped considerably to assess the quality of a topic model. Also, we visualized how the documents of the test set were classified and what their similarities in terms of topic weights and most dominant topic were. This method allowed us to gain plenty of insights into a specific topic model, and we could easily see differences between the two models in terms of their classifications of test documents. LOTClass' predictions proved to be sensible as few words were sufficient to provoke the assignment of a document to a topic. Furthermore, the problematic persists as most topics contained only weights for a limited number of topics, as often a topic could not be detected at all.

Differences in the contextualized, pre-trained sentence embedding models had the largest effect on the multilingual evaluation measures of CTM. Progress in that area will benefit the applicability of trained topic models on documents of other languages that may not have enough domain specific text documents to allow the training of a qualitative satisfactory topic model.

We have examined some of the limitations of CTM and the method of autoencoded variational inference for topic models. We observed that we could not reduce loss by a significant amount after a few epochs and must therefore evaluate the usefulness of the loss function utilized and the assumptions regarding the prior distribution of the latent variables, that is tried to approximate.

References

N. Aletras and M. Stevenson. Evaluating topic coherence using distributional semantics. In K. Erk and A. Koller, editors, *Proceedings of the 10th International Conference on Computational Semantics, IWCS 2013, March 19-22, 2013, University of Potsdam, Potsdam, Germany*, pages 13–22. The Association for Computer Linguistics, 2013. URL https://aclanthology.org/W13-0102/.

S. Bano and N. Khan. A survey of data clustering methods. *International Journal of Advanced Science and Technology*, 113, 04 2018. doi: 10.14257/ijast.2018.113.14.

F. Bianchi, S. Terragni, and D. Hovy. Pre-training is a hot topic: Contextualized document embeddings improve topic coherence. In C. Zong, F. Xia, W. Li, and R. Navigli, editors, *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, ACL/IJCNLP 2021, (Volume 2: Short Papers), Virtual Event, August 1-6, 2021*, pages 759–766. Association for Computational Linguistics, 2021a. doi: 10.18653/v1/2021.acl-short.96. URL https://doi.org/10.18653/v1/2021.acl-short.96.

F. Bianchi, S. Terragni, D. Hovy, D. Nozza, and E. Fersini. Cross-lingual contextualized topic models with zero-shot learning. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 1676–1683, Online, Apr. 2021b. Association for Computational Linguistics. doi: 10.18653/v1/2021.eacl-main.143. URL https://aclanthology.org/2021.eacl-main.143.

D. M. Blei. Introduction to probabilistic topic models. *Communications of the ACM*, 2012.

D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent dirichlet allocation. *Journal of Machine Learning Research*, 2003.

D. M. Blei, A. Kucukelbir, and J. D. McAuliffe. Variational inference: A review for

statisticians. *CoRR*, abs/1601.00670, 2016. URL http://arxiv.org/abs/1601.00670.

M. Brümmer, M. Dojchinovski, and S. Hellmann. Dbpedia abstracts: A large-scale, open, multilingual NLP training corpus. In N. Calzolari, K. Choukri, T. Declerck, S. Goggi, M. Grobelnik, B. Maegaard, J. Mariani, H. Mazo, A. Moreno, J. Odijk, and S. Piperidis, editors, *Proceedings of the Tenth International Conference on Language Resources and Evaluation LREC 2016, Portorož, Slovenia, May 23-28, 2016*. European Language Resources Association (ELRA), 2016. URL http://www.lrec-conf.org/proceedings/lrec2016/summaries/895.html.

J. Chuang, C. D. Manning, and J. Heer. Termite: Visualization techniques for assessing textual topic models. In *Advanced Visual Interfaces*, 2012. URL http://vis.stanford.edu/papers/termite.

A. Conneau and G. Lample. Cross-lingual language model pretraining. In H. M. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. B. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 7057–7067, 2019. URL https://proceedings.neurips.cc/paper/2019/hash/c04c19c2c2474dbf5f7ac4372c5b9af1-Abstract.html.

A. Conneau, G. Lample, R. Rinott, A. Williams, S. R. Bowman, H. Schwenk, and V. Stoyanov. XNLI: evaluating cross-lingual sentence representations. *CoRR*, abs/1809.05053, 2018. URL http://arxiv.org/abs/1809.05053.

J. Devlin, M. Chang, K. Lee, and K. Toutanova. BERT: pre-training of deep bidirectional transformers for language understanding. In J. Burstein, C. Doran, and T. Solorio, editors, *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 4171–4186. Association for Computational Linguistics, 2019. doi: 10.18653/v1/n19-1423. URL https://doi.org/10.18653/v1/n19-1423.

I. Goodfellow, Y. Bengio, and A. Courville. *Deep Learning*. MIT Press, 2016. http://www.deeplearningbook.org.

D. Kingma and M. Welling. *Auto-encoding variational Bayes*. 2014.

D. P. Kingma and M. Welling. An introduction to variational autoencoders. *Found. Trends Mach. Learn.*, 12(4):307–392, 2019. doi: 10.1561/2200000056. URL https://doi.org/10.1561/2200000056.

P. Koehn. *Neural machine translation*. Cambridge University Press, New York, first edit edition, 2020. ISBN 9781108497329.

Y. Liu, 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. *CoRR*, abs/1907.11692, 2019. URL http://arxiv.org/abs/1907.11692.

Y. Meng, Y. Zhang, J. Huang, C. Xiong, H. Ji, C. Zhang, and J. Han. Text Classification Using Label Names Only: A Language Model Self-Training Approach. pages 9006–9017, 2020. doi: 10.18653/v1/2020.emnlp-main.724.

T. Mikolov, K. Chen, G. Corrado, and J. Dean. Efficient estimation of word representations in vector space. In Y. Bengio and Y. LeCun, editors, *1st International Conference on Learning Representations, ICLR 2013, Scottsdale, Arizona, USA, May 2-4, 2013, Workshop Track Proceedings*, 2013. URL http://arxiv.org/abs/1301.3781.

T. Mitchell. *Machine Learning*. McGraw-Hill, New York, 1997. ISBN 0070428077.

H. Mohebbi, A. Modarressi, and M. T. Pilehvar. Exploring the role of bert token representations to explain sentence probing results, 2021.

S. Raaijmakers. *Deep Learning for Natural Language Processing*. Manning Publications Company, City, 2019. ISBN 9781617295447.

N. Reimers and I. Gurevych. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 11 2019. URL http://arxiv.org/abs/1908.10084.

N. Reimers and I. Gurevych. Making monolingual sentence embeddings multilingual using knowledge distillation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 11 2020. URL https://arxiv.org/abs/2004.09813.

D. Rezende, S. Mohamed, and D. Wierstra. *Stochastic backpropagation and approximate inference in deep generative models*. 2014.

M. Röder, A. Both, and A. Hinneburg. Exploring the space of topic coherence measures. In X. Cheng, H. Li, E. Gabrilovich, and J. Tang, editors, *Proceedings of the Eighth ACM International Conference on Web Search and Data Mining, WSDM 2015, Shanghai, China, February 2-6, 2015*, pages 399–408. ACM, 2015. doi: 10.1145/2684822.2685324. URL https://doi.org/10.1145/2684822.2685324.

M. Z. Rodriguez, C. H. Comin, D. Casanova, O. M. Bruno, D. R. Amancio, L. d. F. Costa, and F. A. Rodrigues. Clustering algorithms: A comparative approach. *PLOS ONE*, 14(1):1–34, 2019. doi: 10.1371/journal.pone.0210236. URL https://doi.org/10.1371/journal.pone.0210236.

M. Rosen-Zvi, T. Griths, M. Steyvers, and P. Smith. The author-topic model for authors and documents. *Proceedings of the 20th Conference on Uncertainty inArticial Intelligence*, 2004.

F. Rosner, A. Hinneburg, M. Röder, M. Nettling, and A. Both. Evaluating topic coherence measures. *CoRR*, abs/1403.6397, 2014. URL http://arxiv.org/abs/1403.6397.

V. Sanh, L. Debut, J. Chaumond, and T. Wolf. Distilbert, a distilled version of BERT: smaller, faster, cheaper and lighter. *CoRR*, abs/1910.01108, 2019. URL http://arxiv.org/abs/1910.01108.

R. Sennrich, B. Haddow, and A. Birch. Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016, August 7-12, 2016, Berlin, Germany, Volume 1: Long Papers*. The Association for Computer Linguistics, 2016. doi: 10.18653/v1/p16-1162. URL https://doi.org/10.18653/v1/p16-1162.

C. Sievert and K. Shirley. LDAvis: A method for visualizing and interpreting topics. In *Proceedings of the Workshop on Interactive Language Learning, Visualization, and Interfaces*, pages 63–70, Baltimore, Maryland, USA, June 2014. Association for Computational Linguistics. doi: 10.3115/v1/W14-3110. URL https://aclanthology.org/W14-3110.

P. Sitikhu, K. Pahi, P. Thapa, and S. Shakya. A comparison of semantic similarity methods for maximum human interpretability. *CoRR*, abs/1910.09129, 2019. URL http://arxiv.org/abs/1910.09129.

A. Srivastava and C. Sutton. Autoencoding variational inference for topic models, 2017.

M. Steyvers and T. Griths. *Probabilistic topic models*. 2006.

J. Tzeng, H. H.-S. Lu, and W.-H. Li. Multidimensional scaling for large genomic data sets. 9(1), Apr. 2008. doi: 10.1186/1471-2105-9-179. URL https://doi.org/10.1186/1471-2105-9-179.

L. van der Maaten and G. Hinton. Visualizing data using t-sne. *Journal of Machine Learning Research*, 9(86):2579–2605, 2008. URL http://jmlr.org/papers/v9/vandermaaten08a.html.

A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin. Attention is all you need. In I. Guyon, U. von Luxburg, S. Bengio, H. M. Wallach, R. Fergus, S. V. N. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 5998–6008, 2017. URL https://proceedings.neurips.cc/paper/2017/hash/3f5ee243547dee91fbd053c1c4a845aa-Abstract.html.

M. Wattenberg, F. Viégas, and I. Johnson. How to use t-sne effectively. *Distill*, 2016. doi: 10.23915/distill.00002. URL http://distill.pub/2016/misread-tsne.

T. White. Sampling generative networks: Notes on a few effective techniques. *CoRR*, abs/1609.04468, 2016. URL http://arxiv.org/abs/1609.04468.

Y. Wu, M. Schuster, Z. Chen, Q. V. Le, M. Norouzi, W. Macherey, M. Krikun, Y. Cao, Q. Gao, K. Macherey, J. Klingner, A. Shah, M. Johnson, X. Liu, L. Kaiser, S. Gouws, Y. Kato, T. Kudo, H. Kazawa, K. Stevens, G. Kurian, N. Patil, W. Wang, C. Young, J. Smith, J. Riesa, A. Rudnick, O. Vinyals, G. Corrado, M. Hughes, and J. Dean. Google's neural machine translation system: Bridging the gap between human and machine translation. *CoRR*, abs/1609.08144, 2016. URL http://arxiv.org/abs/1609.08144.

J. Xie, R. Girshick, and A. Farhadi. Unsupervised deep embedding for clustering analysis. In *Proceedings of the 33rd International Conference on International Conference on Machine Learning - Volume 48*, ICML'16, page 478–487. JMLR.org, 2016.

X. Zhang, J. J. Zhao, and Y. LeCun. Character-level convolutional networks for text classification. In *NIPS*, 2015.

H. Zhao, D. Phung, V. Huynh, Y. Jin, L. Du, and W. L. Buntine. Topic modelling meets deep neural networks: A survey. *CoRR*, abs/2103.00498, 2021. URL `https://arxiv.org/abs/2103.00498`.

A Additional Material

A.0.1 Topic Overviews

Topic ID	Subject	top-10 most important words
1	Geographics	'island', 'river', 'mountain', 'lake', 'islands', 'mount', 'survey', 'peak', 'mountains', 'glacier'
2	directions	'poland', 'west', 'within', 'approximately', 'east', 'capital', 'regional', 'north', 'kilometres'
3	Buildings	'building', 'church', 'built', 'grade', 'house', 'school', 'parish', 'style', 'listed', 'england'
4	U.S. sport	american', 'high', 'school', 'college', 'baseball', 'head', 'nfl', 'university', 'state', 'texas'
5	Olympics, sport	silver', 'summer', 'gold', 'russian', 'bronze', 'medal', 'olympics', 'world', 'winter', 'olympic'
6	movies	'film', 'directed', 'stars', 'produced', 'films', 'drama', 'roles', 'comedy', 'written', 'role'
7	military	'war', 'air', 'force', 'army', 'military', 'ship', 'navy', 'forces', 'royal', 'naval'
8	sports	'played', 'made', 'league', 'cricket', 'born', 'player', 'club', 'career', 'first', 'professional'
9	Animals	'native', 'plant', 'name', 'common', 'fish', 'commonly', 'spoken', 'tree', 'plants', 'red'
10	Arts	'art', 'writer', 'work', 'works', 'born', 'worked', 'artist', 'known', 'award', 'poetry'
11	United States	'county', 'states', 'national', 'united', 'places', 'state', 'historic', 'register', 'complete', 'list'
12	digital	'software', 'system', 'developed', 'mobile', 'systems', 'available', 'engine', 'data', 'design'
13	news	'company', 'radio', 'owned', 'channel', 'broadcasting', 'news', 'founded', 'inc', 'largest'
14	racing	'held', 'racing', 'race', 'tour', 'races', 'event', 'car', 'tennis', 'world', 'championship',
15	politics	'party', 'government', 'act', 'members', 'president', 'election', 'elections', 'council', 'vote'
16	colours	'brown', 'grey', 'white', 'family', 'found', 'costa', 'black', 'wing', 'orange', 'dark'
17	research	'university', 'research', 'medical', 'science', 'institute', 'education', 'sciences', 'society'
18	United states	'cleveland', 'baltimore', 'bell', 'davis', 'hamilton', 'taylor', 'howard', 'seattle', 'puerto', 'mass'
19	championships	season', 'club', 'league', 'division', 'football', 'team', 'stadium', 'teams', 'cup', 'play'
20	Television	series', 'game', 'show', 'published', 'book', 'comic', 'novel', 'comics', 'character', 'aired'
21	Music	'band', 'album', 'music', 'rock', 'records', 'released', 'studio', 'label', 'song', 'pop'
22	Medieval	king', 'prince', 'greek', 'rome', 'roman', 'iii', 'saint', 'duke', 'holy', 'maria'
23	Politics	served', 'member', 'politician', 'elected', 'educated', 'john', 'born', 'canadian', 'william'
24	railways	'station', 'road', 'line', 'railway', 'bridge', 'train', 'trains', 'rail', 'highway', 'metro'
25	Village	'town', 'iran', 'population', 'province', 'persian', 'census', 'municipality', 'district', 'city', 'rural'

Table 8: Manual Evaluation of CTM topic quality

A.0.2 LOTClass Category Vocabulary

0	aristocracy	prince', 'king', 'queen', 'kings', 'saint', 'queens', 'royal', 'princess', 'konig', 'roi', 'rey', 'ruler', 'koning', 'san', 'kong', 'knight', 'raja', 'lord', 'saints', 'father', 'sant', 'santo', 'crown', 'god', 'roman', 'monarch', 'emperor', 'kingdom', 'latin', 'santa', 'sainte', 'romano', 'romana', 'sankt', 'roma', 'single', 'lady', 'romans', 'sint', 'greek', 'sir', 'christian', 'prinz', 'szent', 'rex', 'sent', 'sao', 'romani', 'william', 'james', 'prins', 'sancti', 'christ', 'princes', 'marie', 'human', 'principe', 'pont', 'kral', 'catholic', 'sac', 'lat'
1	politics	'government', 'parliament', 'member', 'candidate', 'members', 'party', 'membro', 'ministry', 'cabinet', 'congress', 'lid', 'miembro', 'representative', 'parties', 'mitglied', 'membership', 'election', 'speaker', 'movement', 'minister', 'secretary', 'chairman', 'governments', 'fellow', 'elected', 'partido', 'govern', 'head', 'senator', 'front', 'electoral', 'membre', 'elections', 'chair', 'mp', 'gouvernement', 'regular', 'gobierno', 'membru', 'deputy', 'power', 'governo', 'rally', 'parti', 'charge', 'administration', 'authorities', 'clen', 'wahl'
2	contests	'year', 'anno', 'week', 'annee', 'jaar', 'ano', 'jahr', 'held', 'hold', 'holding', 'holds', 'वरष', 'contested', 'conducted', 'given', 'open', 'organised', 'race', 'took', 'races', 'performed', 'gehouden', 'staged', 'january', 'months', 'drawn', 'racing', 'roku', 'awarded', 'running', 'entry', 'год', 'celebrated', 'rok', 'occurred'
3	TV, Music	tv', 'radio', 'television', 'music', 'broadcasting', 'news', 'show', 'tele', 'broadcast', 'televisio', 'musik', 'musica', 'televisie', 'televisi', 'musique', 'muzik', 'televisao', 'radios', 'entertainment', 'musikk', 'muziek', 'televisivo', 'fm', 'audio', 'televizyon', 'televisiva', 'sound', 'art', 'songs', 'classical', 'composition', 'jazz', 'bass', 'bbc', 'тепе', 'talk', 'communications', 'magazine', 'ser', 'radyo', 'televizni', 'pop', 'радио', '□□□□□□□□□□', 'opera', 'compositions', 'weekly', 'מוזיקה']
4	Film	'film', 'films', 'cinema', 'filme', 'phim', 'filmi', 'filmu', 'фильм', 'filem', 'филм', 'filmen', 'movies', 'pelicula', 'فيلم', 'filmy', 'filmmaker', 'directed', 'filma', 'documentary', '□□□□□', 'remake', 'ταινια', 'кино', 'cine', 'фильма', 'picture', 'spielfilm', 'фильме', 'pellicola', 'oscar', 'filmes', 'screen', 'awards', 'фільм', 'award', 'filmova', 'prize', 'honor', 'medal', 'pictures', 'directing', 'due', 'mu', 'script', 'shot', 'scored', 'directors', 'mention', 'filmed', 'kino', 'camera', 'dirigido', 'festival', 'ordered', 'diretto', 'dirigida', 'regisseur
5	IT	system', 'sistem', 'sistema', 'systeme', 'development', 'software', 'developed', 'sistemi', 'systeem', 'developing', 'develop', 'scheme', 'sistemas', 'model', 'systemet', 'method', 'originated', 'систему', 'ontwikkeld', 'server', 'adapted', 'computer', 'систем', 'database', 'системы', 'entwickelt', 'classification', 'advanced', 'systemu', 'available', 'type', 'accessible', 'possible', 'online', 'pc', 'developpe', 'desenvolvido', 'desarrollado', 'sa', 'solution', 'form', 'linux', 'free', 'operator', 'problem', 'conceived',

Figure 12: LOTClass' derived category vocabulary on the W2 dataset. Part 1.

6	Nature, China	china', 'chinese', 'native', 'chinois', 'common', 'chines', 'cinese', 'russian', 'chine', 'chinoise', 'kinai', 'japanese', 'китаи', 'kinesisk', 'indigenous', 'tionghoa', 'rare', 'traditional', 'han', 'tibetan', 'natives', 'widespread', 'chinesischen', 'kinesiske', 'nativa', 'plant', 'simplified', 'plants', 'nativo', 'unique', 'typical', 'comun', 'planta', 'commonly', 'portuguese', 'pinyin', 'mandarin', 'kinesiska', 'چینی', 'full', 'simple', 'cin', 'pure', 'herb', 'broad', 'commun', 'coastal', 'modern', 'plante', 'perennial', 'asia', 'qing', 'india', 'asie', 'eurasia'
7	railway	line', 'station', 'railway', 'train', 'rail', 'railroad', 'trains', 'stations', 'estacion', 'route', 'lines', 'railways', 'stazione', 'estacao', 'stop', 'linea', 'junction', 'terminus', 'base', 'depot', 'zone', 'gare', 'terminal', 'ligne', 'linie', 'metro', 'link', 'position', 'gauge', 'stasjon', 'coast', 'circle', 'eisenbahn', 'станция', 'tracks', 'halt', 'border', 'rails', 'frequency', 'ferrocarril', 'express', 'ferrovia', 'ferroviaria', 'bahn', 'lijn', 'connection', 'subway', 'yard', 'stasiun', 'stationen
8	law	serving', 'served', 'serve', 'serves', 'worked', 'court', 'lived', 'law', 'acted', 'servit', 'represented', 'justice', 'managed', 'trial', 'resigned', 'courts', 'became', 'spent', 'helped', 'appointed', 'laws', 'cases', 'lawyer', 'legal', 'judicial', 'judge', 'bar', 'assisted', 'terms', 'retired', 'servi', 'later', 'legislation', 'procedure', 'practice', 'tribunal', 'covered', 'succeeded', 'bench', 'appeal', 'служил', 'dispute', 'briefly', 'remained', 'jurist', 'continued', 'policy', 'cour', 'criminal', 'judges', 'circuit', 'commission', 'jury', 'functions', 'received', 'sat', 'governed', 'survived', 'attorney', 'existed', 'function', 'lawyers'
9	numbers	'first', 'third', 'primer', 'last', 'premier', 'primera', 'eerste', 'primeira', 'ersten', 'ilk', 'elso', 'erste', 'prime', 'første', 'premiere', 'prima', 'primeiro', 'prvi', 'fourth', 'prvni', 'earliest', 'next', 'inaugural', 'initial', 'первыи', 'fifth', 'sixth', 'primo', 'largest', 'opening', 'forsta', 'early', 'made', 'initially', 'club', 'make', 'making', 'best', 'top', 'clubs', 'makes', 'klub', 'original', 'clube', 'prva', 'seventh', 'earste', 'long', 'final', 'following', 'erster', 'premieres', 'official', 'began', 'premiers', 'клуб
10	business	'company', 'companies', 'corporation', 'co', 'compania', 'organization', 'compagnie', 'entity', 'manufacturer', 'founded', 'services', 'works', 'компания', 'fundada', 'gegrundet', 'production', 'компании', 'fondata', 'incorporated', 'fundado', 'grunnlagt', 'opgericht', 'organisation', 'основана', 'compagnia', 'companhia', 'fondee', 'factory', 'operations', 'agency', 'основан', 'limited', 'основано', 'fondato', 'chartered', 'grundlagt', 'organizations', 'found', 'założona', 'acquired', 'operation', 'grundete', 'giant', 'chain', 'założony', 'unternehmen', 'products', 'corp', 'industry', 'owned', 'product'
11	Music	'album', 'released', 'band', 'ep', 'cd', 'albums', 'records', 'lp', 'disco', 'label', 'albumu', 'albumet', 'releasing', 'албум', 'альбом', 'albume', 'recordings', 'αλμπουμ', 'albumi', 'signed', 'releases', 'studioalbum', 'uitgebracht', 'albuma', 'κυκλοφορησε', 'artist', 'bands', 'rock', 'material', 'альбома', 'banda', 'האלבום', 'выпущен', 'bande', 'extended', 'dvd', 'tape', 'opus', 'slapptes', 'appearance', 'unit', 'supported', 'studio', 'lancado', 'uitgegeven', 'gruppe', 'udgivet', 'dirilis
12	geography	'south', 'west', 'east', 'ost', 'southeast', 'northeast', 'oost', 'sud', 'southwest', 'northwest', 'sudwest', 'øst', 'zuid', 'sooth', 'ouest', 'est', 'eastern', 'nord', 'barat', 'דרום', 'suid', 'occidental', 'vest', 'syd', 'far', 'oster', 'km', 'sur', 'timur', 'mi', 'si', 'mil', 'oriental', 'miles', 'sq', 'ми', 'sør', 'mile', 'სამხრეთ', 'juzno', 'habagatan', 'away', 'oest', 'ahead', 'בדרום', 'mph', 'kilometres', 'guney', 'zapad', 'juzna', 'opposite', 'מערב', 'sub', 'near', 'mid', 'pam', 'юго', 'kelet', 'poli', 'sw', 'wes'
13	war	'war', 'army', 'military', 'guerra', 'air', 'wars', 'guerre', 'forces', 'conflict', 'battle', 'armed', 'воины', 'force', 'ist', 'command', 'defense', 'воина', 'navy', 'armee', 'aviation', 'corps', 'raf', 'krieg', 'fighting', 'defence', 'infantry', 'aire', 'gerra', 'soldiers', 'troops', 'soldier', 'militar', 'luft', 'sky', 'oorlog', 'aer', 'aero', 'flying', 'militaire', 'воине', 'jet', 'airs', 'civilian', 'naval', 'aircraft', 'armies', 'wir', 'ejercito', 'reconstruction', 'warfare', 'medical', 'flight', 'space', 'artillery', 'wall', 'guard'
14	buildings	national', 'building', 'nacional', 'nationale', 'built', 'house', 'nemzeti', 'naitional', 'nasional', 'construction', 'property', 'build', 'nazionale', 'houses', 'constructed', 'designated', 'complex', 'reserve', 'buildings', 'nationaal', 'narodni', 'named', 'erected', 'construit', 'wing', 'rebuilt', 'construido', 'placed', 'listed', 'casa', 'register', 'added', 'metropolitan', 'residence', 'construida', 'tower', 'library', 'edificio', 'personal', 'nationally', 'nat', 'gebaut', 'erbaut', 'национальныи', 'построен', 'frame', 'homes', 'bygget', 'национальноı', 'gebouwd'
15	nationalities	'born', 'født', 'נולד', 'sinh', 'birth', 'american', 'ne', 'geboren', 'america', 'americans', 'canadian', 'lahir', 'nascut', 'died', 'amerikan', 'родился', 'texas', 'נולדה', 'americain', 'professional', 'szuletett', 'nacido', 'amerika', 'mexican', 'estadounidense', 'fodd', 'americano', 'married', 'california', 'født', 'us', 'amerikai', 'amer', 'may', 'december', 'americana', 'amerikansk', 'nada', 'rođen', 'americaine', 'amerikaanse', 'march', 'אמריקאי', 'italian', 'nee', 'рођен', 'amerikaans', 'august', 'back', 'जनम', 'род'

Figure 13: LOTClass' derived category vocabulary on the W2 dataset. Part 2.

16	books	'series', 'book', 'books', 'serie', 'publication', 'publishing', 'novel', 'published', 'story', 'sorozat', 'illustrated', 'publish', 'memoir', 'essay', 'printed', 'novels', 'libro', 'read', 'buch', 'publicado', 'publicada', 'reeks', 'boek', 'fiction', 'publicat', 'reported', 'trilogy', 'literature', 'episodes', 'collected', 'книга', 'strip', 'title', 'stories', 'category', 'volumes', 'soap', 'writing', 'серии', 'livro', 'книги', 'programs
17	U.S. history	'jones', 'taylor', 'johnson', 'carter', 'wright', 'washington', 'harris', 'hughes', 'rogers', 'clark', 'evans', 'houston', 'thomas', 'davis', 'williams', 'howard', 'baker', 'baltimore', 'bell', 'henry', 'philadelphia', 'francis', 'stewart', 'maryland', 'wilson', 'miami', 'lewis', 'cleveland', 'bells', 'chicago', 'phillips', 'adams', 'hopkins', 'wood', 'tyler', 'young', 'scott', 'watts', 'thompson', 'powell', 'bull', 'david', 'cincinnati', 'davies', 'robert', 'turner', 'moore', 'md', 'burton', 'seymour', 'london', 'jenkins', 'salisbury', 'delaware', 'kelly', 'roberts', 'walker', 'clarke', 'jacksonville', 'jersey', 'berkeley', 'gordon', 'edwards', 'gardner', 'pittsburgh', 'carpenter', 'hayward', 'thomson', 'jackson', 'fire', 'columbus', 'mitchell', 'watkins', 'wheeler', 'preston', 'ball', 'meyers', 'rhodes', 'alexander', 'athens'
18	sports	'world', 'warld', 'welt', 'mundial', 'dunya', 'wereld', 'dunia', 'worlds', 'mondo', 'maailman', 'mundo', 'global', 'mon', 'olympic', 'мировои', 'silver', 'summer', 'svet', 'pasaules', 'mondiale', 'maailma', 'mondial', 'junior', 'monde', 'gold', 'worldwide', 'games', 'commonwealth', 'planet', 'nations', 'verden', 'olympics', 'gulf', 'verdens', 'vilag', 'bronze', 'sommer', 'copper', 'pen', 'sports', 'web', 'pacific', 'உலக', 'youth', 'globe', '□□□□', 'silber', 'verano', 'argent', 'seven', 'intercontinental', 'europa', 'universe', 'zomer', 'mundi', 'goud', 'univers'
19	science	'university', 'institute', 'universities', 'universitat', 'universite', 'univ', 'universidad', 'universidade', 'uni', 'universiteit', 'universita', 'research', 'faculty', 'trinity', 'centre', 'future', 'scientist', 'universiti', 'varsity', 'science', 'universitesi', 'universida', 'universitas', 'northwestern', 'studies', 'hochschule', 'universitet', 'seminary', 'institut', 'associate', 'museum', 'sciences', 'academic', 'universitatea', 'study', 'professor', 'exchange', 'университета', 'университете', 'institutes', 'tech', 'scholar', 'professors', 'researcher', 'campus', 'graduate', 'teacher', 'profesor', 'lecturer', 'professori', 'instituto', 'scientific
20	school	'state', 'county', 'high', 'school', 'counties', 'schools', 'ecole', 'escuela', 'megye', 'schule', 'escola', 'etat', 'estado', 'stat', 'condado', 'community', 'secondary', 'scuola', 'کاونتی', 'elementary', 'школы', 'low', 'staat', 'higher', 'школе', 'review', 'fylke', 'sekolah', 'pa', 'sr', 'educational', 'estat', 'stato', 'sc'
21	sports	'season', 'team', 'teams', 'division', 'league', 'game', 'saison', 'temporada', 'squad', 'player', 'seizoen', 'sezon', 'stagione', 'sezonu', 'conference', 'mannschaft', 'star', 'сезон', 'equipo', '□□□□□', 'crew', 'equipe', 'сезона', 'divisions', 'football', 'championship', 'men', 'сезоне', 'laget', 'playing', 'body', 'sesongen', 'draft', 'sezone', 'divisao', 'tempada', 'sæson', 'divisio'
22	nature	'lake', 'river', 'bay', 'creek', 'rivers', 'rivier', 'brook', 'riviere', 'mountain', 'rio', 'island', 'fluss', 'fiume', 'peninsula', 'islands', 'ile', 'isla', 'park', 'insel', 'eiland', 'archipelago', 'lakes', 'peak', 'ilha', 'islanders', 'mount', 'street', 'isola', 'waters', 'stream', 'little', 'mountains', 'beach', 'lac', 'atoll', 'insula', 'lago', 'mainland', 'ocean', 'pulau', 'reservoir', 'riu', 'mar', 'confluence', 'current', 'former', 'illa', 'dam', 'mt', 'flow', 'canal'
23	city	'town', 'population', 'municipality', 'province', 'village', 'commune', 'settlement', 'capital', 'towns', 'colony', 'populations', 'figure', 'poblacion', 'hamlet', 'statistics', 'numbers', 'urban', 'toun', 'density', 'rate', 'municipalities', 'villa', 'ceety', 'situation', 'inhabitants', 'cities', 'provincia', 'populace', 'provincie', 'municipio', 'provinces', 'provinz', 'popolazione', 'comune', 'information', 'citta', 'capacity', 'cidade', 'ville', 'poboacion', 'metropolis', 'poblacio', 'populacao', 'data', 'gemeente', 'administrative', 'kommune', 'bevolkerung'
24	family	family', 'familia', 'familie', 'families', 'famille', 'famiglia', 'black', 'white', 'gray', 'brown', 'dynasty', 'grey', 'семеиство', 'familiar', 'pink', 'csalad', 'household', 'subfamily', 'purple', 'dark', 'class', 'colour', 'семеиства', 'clan', 'order', 'pale', 'familjen', 'orange', 'rodziny', 'familien', 'famili', 'brothers', 'oil', 'estate', 'coloured', 'braun', 'tribe', 'brun', 'van', 'colored', 'rose', 'οικογενεια', 'din', 'familias', 'relatives', 'seima', 'browns

Figure 14: LOTClass' derived category vocabulary on the W2 dataset. Part 3.

A.0.3 Additional Material to Chapter 6

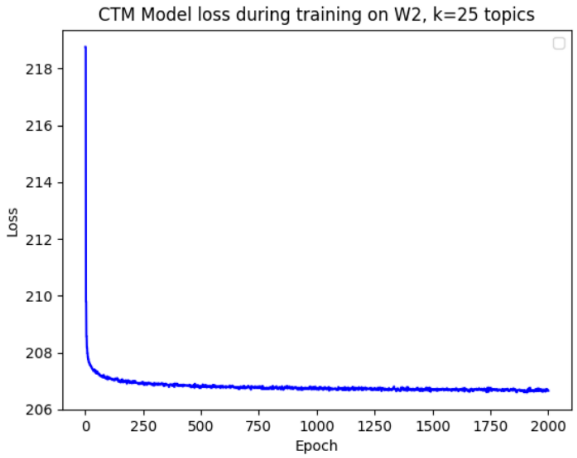

Figure 15: Train loss during 2000 epochs of training a CTM model is changing only marginally after the first 20 epochs.

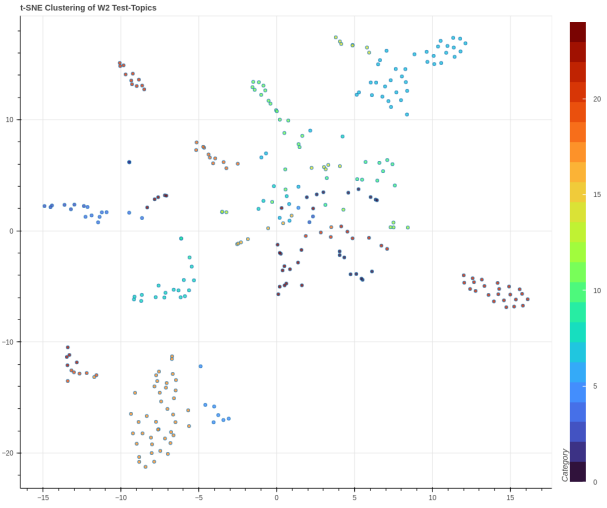

Figure 16: CTM's prediction of topic weights on the **English** test documents. Colors represent the most dominant topic, positions are obtained using t-SNE dimensionality reduction, therefore positions are to be interpreted relatively.

A.0.4 Labels Used as Supervision for LOTClass

supervision labels used when providing at most 5 labels per category	when providing one label per cateogry
king prince saint lord roman	king
party member election minister government	government
held race racing year calendar	race
radio music television show tv	radio
film directed stars films award	film
developed software available engine system	software
native common chinese plant asia	asia
station line rail railway trains	railway
served court judge law	law
club cricket made first	club
company organization services founded companies	company
album released band rock records	band
east west mi poland south	east
war air army force military	war
built house building national listed	building
American baseball professional born	baseball
series book published novel comic	book
taylor bell davis baltimore howard	baltimore
silver gold summer world olympic	Olympic
university professor studies institute science	university
school county high state community	county
team head season division conference	team
mountain river island mount lake	mountain
iran province town population municipality	population
brown white grey family	brown

Table 9: Labels that were used as weak-supervision input for LOTClass in the cases of providing at most 5 labels or only one label. The case of providing 20 labels was omitted here as most of them can be found in Figures 12, 13, 14

A.0.5 CTM Detailed Results Table: 3

20 epochs NPMI SCORE: 0.20 (using 10 words per topic)
Comparing EN with EN translation from GER:
Centroid Distance: 0.83
Average KL Divergence: 0.28
MATCHES: 224 of 300. In %: 75

Comparing GER with EN translation from GER:
Centroid Distance: 0.87
Average KL Divergence: 0.21
MATCHES: 240 of 300. In %: 80

Comparing EN with PORT:
Centroid Distance: 0.82
KL Divergence: 0.26
MATCHES: 221 of 300. In %: 74

Comparing EN with GER:
Centroid Distance: 0.80
KL Divergence: 0.29
MATCHES: 212 of 300. In %: 71

Comparing EN with IT:
Centroid Distance: 0.81
KL Divergence: 0.28
MATCHES: 213 of 300. In %: 71

Comparing EN with FR:
Centroid Distance: 0.83
KL Divergence: 0.28
MATCHES: 222 of 300. In %: 0.74

100 epochs NPMI SCORE: 0.20 (using 10 words per topic)
Comparing EN with PORT:
Centroid Distance: 0.82

KL Divergence: 0.30
MATCHES: 226 of 300. In %: 75

Comparing EN with GER:
Centroid Distance: 0.79
KL Divergence: 0.35
MATCHES: 216 of 300. In %: 72

Comparing EN with IT:
Centroid Distance: 0.79
KL Divergence: 0.31
MATCHES: 214 of 300. In %: 71

Comparing EN with FR:
Centroid Distance: 0.79
KL Divergence: 0.33
MATCHES: 212 of 300. In %: 71

300 epochs NPMI SCORE: 0.19 (using 10 words per topic)
Comparing EN with PORT:
Centroid Distance: 0.82
KL Divergence: 0.31
MATCHES: 225 of 300. In %: 75

Comparing EN with GER:
Centroid Distance: 0.78
KL Divergence: 0.34
MATCHES: 204 of 300. In %: 68

Comparing EN with IT:
Centroid Distance: 0.80
KL Divergence: 0.33
MATCHES: 214 of 300. In %: 71

Comparing EN with FR:

Centroid Distance: 0.78
KL Divergence: 0.35
MATCHES: 208 of 300. In %: 69

500 epochs NPMI SCORE: 0.18 (using 10 words per topic)
Comparing EN with EN translation from GER:
Centroid Distance: 0.77
Average KL Divergence: 0.33
MATCHES: 201 of 300. In %: 67

Comparing GER with EN translation from GER:
Centroid Distance: 0.82
Average KL Divergence: 0.23
MATCHES: 223 of 300. In %: 74

Comparing EN with PORT:
Centroid Distance: 0.79
Average KL Divergence: 0.35
MATCHES: 210 of 300. In %: 70

Comparing EN with GER:
Centroid Distance: 0.77
Average KL Divergence: 0.35
MATCHES: 198 of 300. In %: 66

Comparing EN with IT:
Centroid Distance: 0.77
Average KL Divergence: 0.35
MATCHES: 201 of 300. In %: 67

Comparing EN with FR:
Centroid Distance: 0.77
Average KL Divergence: 0.38
MATCHES: 196 of 300. In %: 65

A.0.6 LOTClass Detailed Results Table 7

	Matches (in %) (n=25)	Centroid Distance (n=25)	KL Divergence (n=25)
German	86	0.895	0.324
French	86	0.900	0.358
Italian	75	0.841	0.740
Portuguese	86	0.894	0.328

Table 10: Comparison metrics between the English test set and the test sets in multiple other languages. LOTClass was used with at most **5** words as weak-supervision input. NPMI score on the training set was 0.123

	Matches (in %) (n=25)	Centroid Distance (n=25)	KL Divergence (n=25)
German	77	0.835	0.509
French	81	0.861	0.467
Italian	74	0.810	0.668
Portuguese	78	0.844	0.472

Table 11: Comparison of similarity between the predicted topic distributions on the different test sets. Weak-supervision input was extended to the whole category vocabulary (at most **20** words per category)that was calculated in an earlier run. NPMI score on the training set was 0.128

	Matches (in %) (n=25)	Centroid Distance (n=25)	KL Divergence (n=25)
German	68	0.760	0.942
French	67	0.748	0.961
Italian	69	0.769	0.963
Portuguese	69	0.764	0.893

Table 12: Comparison of similarity between the predicted topic distributions on the different test sets. Weak-supervision input was reduced to the semantically most general label (**1**) per category. NPMI score on the training set was 0.072

B Default Model Configuration

B.1 LOTClass

LOTClass was used under the standard configuration settings and the parameters that were used on the AG's News dataset, as described in Meng et al. [2020] since W2 has similar attributes. This includes clipping of articles to a maximum length of 200 tokens. Training took place on two GeForce GTX Titan X (12GB) GPUs. Train batch size was set to 32 and evaluation batch size to 128. The model was trained on the MCP step for 3 epochs, and self training was set to 1 epoch. Tokenization of the input text was performed using the integrated tokenizer of the pre-trained BERT model *bert-base-multilingual-uncased*. Category vocabulary size was kept at 100, during MLM the top prediction cutoff was set to 100, and match-threshold for category indicative words was set to 20. The number of gradient accumulation steps was set to 2.

B.2 CTM

Default training duration was set to 20 epochs, dropout was set to 0.2 by default, batch size was set to 64, learning rate to 0.002, and a momentum of 0.99 was used. Training took place on one GeForce GTX Titan X (12GB) using the language model *paraphrase-multilingual-mpnet-base-v2*. Preprocessing was done using the model's built-in tokenizer functionality. Furthermore, for the BoW input representation, stopwords were removed using NLTK, also the *WhiteSpacePreprocessing* method of CTM was used which sets all characters to lowercase and filters out infrequent tokens and punctuation. Vocabulary size was set to 2'000 and the BoW representation was derived using *Scikit-learn*'s *CountVectorizer* method.

Universität Zürich[UZH]

Institut für Computerlinguistik

Selbstständigkeitserklärung

Originalarbeit

Ich erkläre ausdrücklich, dass es sich bei der von mir im Frühjahrs-/Herbst-Semester 20..21.. an der Universität Zürich eingereichten schriftlichen Arbeit mit dem Titel

Zero-Shot Cross-lingual Transfer of the Topic Modeling Task

um eine von mir selbst und ohne unerlaubte Beihilfe sowie in eigenen Worten verfasste Originalarbeit handelt. Sofern es sich dabei um eine Arbeit von mehreren Verfasserinnen oder Verfassern handelt, bestätige ich, dass die entsprechenden Teile der Arbeit korrekt und klar gekennzeichnet und der jeweiligen Autorin oder dem jeweiligen Autor eindeutig zuzuordnen sind.

Ich bestätige überdies, dass die Arbeit als Ganzes oder in Teilen weder bereits einmal zur Abgeltung anderer Studienleistungen an der Universität Zürich oder an einer anderen Universität oder Ausbildungseinrichtung eingereicht worden ist noch inskünftig durch mein Zutun als Abgeltung einer weiteren Studienleistung eingereicht werden wird.

Verwendung von Quellen

Ich erkläre ausdrücklich, dass ich sämtliche in der oben genannten Arbeit enthaltenen Bezüge auf fremde Quellen (einschliesslich Tabellen, Grafiken u. €.) als solche kenntlich gemacht habe. Insbesondere bestätige ich, dass ich ausnahmslos und nach bestem Wissen sowohl bei wärtlich übernommenen Aussagen (Zitaten) als auch bei in eigenen Worten wiedergegebenen Aussagen anderer Autorinnen oder Autoren (Paraphrasen) die Urheberschaft angegeben habe.

Sanktionen

Ich nehme zur Kenntnis, dass eine Arbeit, welche zum Erwerb eines Leistungsnachweises verwendet wird und sich als Plagiat im Sinne des Dokuments *Erläuterung des Begriffs „Plagiat"* erweist, in leichten Fällen zu Notenabzug führt, in schweren Fällen mit Note 1 (eins) ohne Möglichkeit einer Überarbeitung bewertet werden kann und in ganz gravierenden Fällen die entsprechenden rechtlichen und disziplinarischen Konsequenzen nach sich ziehen kann (gemäss §§ 7ff der Disziplinarordnung der Universität Zürich sowie § 36 der Rahmenordnung für das Studium in den Bachelor- und Master-Studiengängen der Philosophischen Fakultät der Universität Zürich).

Ich bestätige mit meiner Unterschrift die Richtigkeit dieser Angaben.

Name: Tinner Vorname: Francesco

Matrikelnummer: 17-709-510

Datum: 30.11.2021 Unterschrift:

Figure 17: Selbständigkeitserklärung